

ROBOT PLANAR PARALELO RRR

**JUAN DIEGO TOVAR MORENO
YORDY HARVEY TORRES ROJAS**

**UNIVERSIDAD PILOTO DE COLOMBIA
FACULTAD DE INGENIERÍA MECATRÓNICA, CUNDINAMARCA
BOGOTÁ
2015**

ROBOT PLANAR PARALELO RRR

**JUAN DIEGO TOVAR MORENO
YORDY HARVEY TORRES ROJAS**

**TRABAJO DE GRADO
como requisito parcial para la obtención del título de
Ingeniero Mecatrónico**

ASESORES

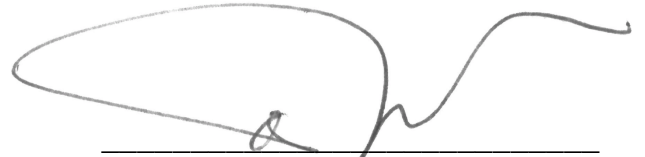
**ING. OSCAR AVILES SANCHEZ. Ph.D
ING. RUBEN DARIO HERNANDEZ BELEÑO. Ph.D(c)**

**UNIVERSIDAD PILOTO DE COLOMBIA
FACULTAD DE INGENIERÍA MECATRÓNICA, CUNDINAMARCA
BOGOTÁ
2015**

Nota de aceptación

El presente trabajo cumple con los requisitos exigidos
para optar al título de Ingeniero Mecatrónico

Prof. Dr. Oscar Avilés



ING. OSCAR AVILES SANCHEZ. Ph.D

ING. JENNIFER CORREDOR. Phd (c)

ING. LUIS CARLOS MENESES. MSc

Bogotá (26, Octubre, 2015)

DEDICATORIA

Este Proyecto va dedicado a nuestras madres, padres, hermanos y hermanas, ya que gracias a ellos, a su apoyo incondicional y a sus valores inculcados en nosotros, tuvimos el entusiasmo de comenzar nuestra carrera y a lo largo de los semestres desarrollarla con total dedicación hasta llegar a este trabajo de grado, que concluye nuestra primera formación como profesionales.

Dedicamos también este proyecto a todos nuestros amigos y amigas quienes con sus conocimientos han aportado a nuestra formación y nos han dado ánimo de seguir adelante, igualmente agradecer a las personas que ya no están con nosotros por diferentes motivos y que contribuyeron con este proceso, no sobra mencionar a las personas que se encuentran en el cielo con nuestro padre celestial, para ellos es este trabajo.

AGRADECIMIENTOS

Queremos agradecer a nuestras familias, ya que gracias a ellos tuvimos la oportunidad de adquirir cada semestre una matrícula, gracias a esos esfuerzos realizados, hoy en día estamos aquí presentando nuestro proyecto de grado.

Así mismo agradecemos a todos los profesores de ingeniería mecatrónica de la Universidad Piloto De Colombia, en especial a nuestros tutores, los ingenieros Oscar Avilés y Rubén Hernández quienes a lo largo de decimo semestre nos guiaron con comprensión y mucha paciencia para poder culminar nuestro proyecto.

CONTENIDO

	pág.
INTRODUCCIÓN.....	16
1. DESCRIPCIÓN DEL SISTEMA	17
1.1 PLANTEAMIENTO DEL PROBLEMA	17
1.2 ANTECEDENTES	17
1.3 DESCRIPCIÓN DEL PROBLEMA.....	22
1.4 FORMULACIÓN DEL PROBLEMA.....	23
1.5 JUSTIFICACIÓN	23
1.6 OBJETIVO GENERAL.....	24
1.7 OBJETIVOS ESPECÍFICOS	24
1.8 ALCANCES Y LIMITACIONES.....	25
2. ROBOTICA.....	26
2.1 ROBOTS INDUSTRIALES	26
2.1.1 Tipos y clasificación de robots industriales.....	26
2.2 ROBOTS TELEOPERADOS Y DE SERVICIO.....	27
2.3 CLASIFICACIÓN DE ROBOTS	27
2.3.1. Estructura mecánica del robot.....	28
2.4 ROBOTS PARALELOS	29
2.4.1 Configuraciones de los robots paralelos.....	30
2.4.1.1 Robots paralelos planares	30
2.4.1.2 Robots paralelos esféricos	32
2.4.1.3 Robots paralelos espaciales	33
3. ROBOT PLANAR PARALELO 3RRR	34
3.1 ANÁLISIS GEOMÉTRICO	34
3.2 ANÁLISIS DE POSICIÓN.....	36
3.3 CINEMÁTICA INVERSA.....	37
3.4 CINEMÁTICA DIRECTA.....	38
3.5 ESPACIO DE TRABAJO	38
3.6 ANÁLISIS JACOBIANO.....	41
3.7 SINGULARIDADES.....	42
3.7.1 Singularidad de cinemática inversa	42
4 DISEÑO MECATRÓNICO	44

4.7	DISEÑO MECANICO	44
4.8	DIMENSIONES DEL PROTOTIPO	45
4.9	COMPONENTES MECANICOS DEL ROBOT	46
4.9.1	ESLABONES <i>(ai)</i>	46
4.9.2	ESLABONES <i>(bi)</i>	47
4.9.3	PLATAFORMA MOVIL	47
4.9.4	BASE CON LOGO UNIVERSIDAD PILOTO DE COLOMBIA	47
4.9.5	BASE SOBRE BASE	48
4.9.6	LATERALES DE LA BASE	49
4.9.7	FRONTALES DE LA BASE	49
4.9.8	ENSAMBLAJE TOTAL DEL ROBOT.....	49
4.10	ANALISIS ESTÁTICO	50
4.11	DISEÑO ELECTRONICO	56
4.11.1	Arduino	57
4.11.2	Servomotor.....	59
4.11.2.1	Servomotor sg-5010 60	.
4.12	INTERFAZ DE USUARIO.....	61
4.12.1	Comunicación matlab, arduino y servomotor	68
4.12.2	Instrucciones para instalar arduino io	69
4.12.3	Manejo de servomotores a traves de arduino io	69
5	ANALISIS DE RESULTADOS	73
5.7	PROTOTIPO REAL	73
5.8	DISTINTOS ESPACIOS DE TRABAJO.....	74
5.9	SERVOMOTOR DE 180 GRADOS	79
5.10	INTERFAZ DE USUARIO.....	84
5.11	PRESICION DEL ROBOT	90
5.12	ANÁLISIS DE COSTOS	92
5.12.1	DISEÑO.....	93
6	TRABAJOS FUTUROS.....	99
7	CONCLUSIONES	100
8	BIBLIOGRAFÍA.....	103
	ANEXO A	105
	ANEXO B.....	111
	ANEXO C.....	116
	ANEXO D.....	121

ANEXO E	123
ANEXO F	166

LISTA DE FIGURAS

pág.

Figura 1. Muñeco musical creado por Jacques De Vauncansos.	18
Figura 2. Muñeca de levas inventada por Henri Maillardert.	18
Figura 3. Supuesto primer diseño de un robot paralelo espacial.	19
Figura 4. Primer robot paralelo.	19
Figura 5. Plataforma de Gough.	20
Figura 6. Robot hexápodo	20
Figura 7. Plataforma de Stewart 1965.	21
Figura 8. Robot IRb6.	21
Figura 9. Máquina cortadora laser DURMA.	22
Figura 10. Prototipo de robot 3RRR.	22
Figura 11. Estación de mando con su robot teleoperado.	27
Figura 12. Robot seriar, híbrido y paralelo	28
Figura 13. Tipos de articulaciones	29
Figura 14. Robot planar	30
Figura 15. Siete combinaciones en serie con su respectiva cadena cinemática cerrada.	32
Figura 16. <i>The agile eye</i> , mecanismo esférico.	32
Figura 17. Robots paralelos espaciales.	33
Figura 18. Robot paralelo planar, 3RRR	34
Figura 19. Manipulador paralelo 3RRR.	35
Figura 20. Cinemática directa en inversa.	36
Figura 21. Espacio de trabajo robot 3RRR	39
Figura 22. Configuración (CAC) y (CAA)	39
Figura 23. Configuración (CCC) y (CCA)	40
Figura 24. Configuración (ACA) y (ACC)	40
Figura 25. Configuración (AAC) y (AAA)	41
FIGURA 26. Arquitectura del robot	44
Figura 27. Partes de un robot planar 3RRR.	45
Figura 28. Corte de acrílico.	46
Figura 29. Diseño eslabón <i>ai</i>	46
Figura 30. Diseño de eslabón <i>bi</i>	47
Figura 31. Diseño de la plataforma móvil.	47
Figura 32. Diseño de la Base con logo de la universidad	48
Figura 33. Diseño de base sobre base	48
Figura 34. Diseño de laterales de la base	49
Figura 35. Diseño de frontales de la base	49
Figura 36. Vista isométrica del robot.	49
Figura 37. Vista explosionada.	50
Figura 38. Tercera cadena cinemática del robot.	50
Figura 39. Esquemático superior base móvil	51
Figura 40. Dimensiones de los eslabones	52
Figura 41. Diagrama de cuerpo libre para eslabones b	53

Figura 42. Diagrama de fuerza cortante y momento flector eslabón 2.....	54
Figura 43. Diagrama de cuerpo libre para eslabones a	54
Figura 44. Diagrama de fuerza cortante y momento flector eslabón 1.....	55
Figura 45. Perfil del eslabón	55
Figura 46. Conexión electrónica	57
Figura 47. Arduino uno	58
Figura 48. Modelado del servo SG-5010	61
Figura 49. Interfaz grafica	61
Figura 50. Ítem 1, Longitudes	62
Figura 51. Ítem 2, Coordenadas	63
Figura 52. Ítem 3, Configuración.....	63
Figura 53. Ítem 4, Gráfica del robot por pasos.....	64
Figura 54. Ítem 5, Área de trabajo	64
Figura 55. Ítem 6, Grafica robot Continua.....	65
Figura 56. Ítem 7, Recorrido del robot	65
Figura 57. Ítem 8, Cinemática Inversa	66
Figura 58. Ítem 9, Botones de acción.	67
Figura 59. Ítem 10, Mensaje de error.....	68
Figura 60. Ubicando carpeta Arduino IO.....	70
Figura 61. Instalación de Arduino IO.....	70
Figura 62. Conexión servomotor-arduino.....	72
Figura 63. Prototipo terminado.....	73
Figura 64. Prototipo con decorado.....	73
Figura 65. Cadenas cinemáticas con longitud de eslabones de 5 cm	75
Figura 66. Cadenas acopladas con eslabones de 5cm	75
Figura 67. Área de trabajo con eslabones de 5 cm.....	76
Figura 68. Prototipo con eslabones de 5 cm en configuración 1	76
Figura 69. Área de trabajo con eslabones de 5cm y 7.5 cm	77
Figura 70. Prototipo con eslabones de 7.5 cm en configuración 1.....	77
Figura 71. Área de trabajo para eslabones de 5 cm 7.5 cm y 10 cm	78
Figura 72. Prototipo con eslabones de 10 cm en configuración 1.....	79
Figura 73. Prototipo en configuración 1	80
Figura 74. Ángulos Obtenidos primera prueba	80
Figura 75. Prototipo en configuración 2	81
Figura 76. Ángulos obtenidos en la primera prueba con configuración 2	81
Figura 77. Prototipo con condicionantes en los ángulos.....	82
Figura 78. Ángulos Obtenidos con condicionantes	83
Figura 79. Ingresando longitudes.....	84
Figura 80. Área de trabajo	84
Figura 81. Coordenadas para simulación	85
Figura 82. Línea Recta con 5 pasos	86
Figura 83. Trayectoria en el área de trabajo de la línea recta de 5 pasos	86
Figura 84. Línea recta de 7 pasos	86
Figura 85. Mensaje de alerta	87
Figura 86. Trayectoria en el área de trabajo de la línea recta de 7 pasos	87
Figura 87. Línea recta de 7 pasos en configuración 2	88
Figura 88. Posición Límite de la línea recta	88
Figura 89. Fin de la figura rectángulo	89

Figura 90. Trayectoria del robot	90
Figura 91. Posición deseada en la interfaz	91
Figura 92. Posición real del robot	91
Figura 93. Sistema de rieles para bombillos	97
Figura 97. Plano coordenado	105
Figura 98. Triangulo	105
Figura 99. Primera cadena cinemática	106
Figura 100. Segunda cadena cinemática	107
Figura 101. Tercera cadena cinemática	108
Figura 102. Jacobiano	121

LISTA DE TABLAS

pág.

Tabla 1. Distintas combinaciones de juntas cinemáticas	31
Tabla 2. Especificaciones técnicas arduino uno	59
Tabla 3. Características servo SG-5010	60
Tabla 4. Porcentaje de precisión.....	91
Tabla 5. Costos de diseño en parte experimental.....	93
Tabla 6, Costos por mes	94
Tabla 7. Costos del corte del acrílico	95
Tabla 8. Costo de elementos principales	95
Tabla 9. Costo total.....	96

GLOSARIO

Articulación mecánica unión entre dos eslabones o sólidos el cual permite el movimiento ya que convergen en un mismo eje o en un punto de apoyo.

Cinemática estudia el movimiento de los cuerpos en condiciones de espacio y tiempo, excluyendo los motivos que lo producen.

Coordenadas este se utiliza para conocer la posición de uno o más puntos en un plano ya sea de dos o de tres dimensiones.

Eector final dispositivo con el que concluye un brazo robótico, diseñado para distintas tareas.

Eslabón pieza mecánica que tiene dos nodos, estos se unen con otros eslabones o articulaciones.

Espacio de trabajo aquella área donde el eector final puede llegar y realizar el respectivo trabajo.

Trayectoria movimiento calculado que realizara un dispositivo mecánico para llegar a un punto.

RESUMEN

El presente documento contiene de forma detallada los conocimientos previos requeridos para comprender la robótica paralela, y la forma en la que a través de la historia se han venido realizando aportes que han contribuido para formar lo que hoy en día se conoce como robótica. En los siguientes capítulos se hará énfasis en el diseño y construcción de un robot paralelo planar de tres grados de libertad rotacionales, para ello se dedicaron algunas secciones del presente documento al modelado matemático del sistema que surge desde el análisis geométrico del movimiento del sistema, continua con la obtención de las ecuaciones que permiten calcular los movimientos articulares de cada una de sus cadenas cinemáticas y la de su efector final, concluyendo la etapa de modelamiento con el análisis de las matrices jacobianas que permiten hallar las velocidades articulares y la de la plataforma móvil que incluye el efector final. El presente documento tiene entre sus epígrafes el diseño mecánico del sistema y la arquitectura de control implementada. En los últimos capítulos se ubica el prototipo diseñado dentro del mercado actual colombiano, realizando una comparación entre funcionalidad, relación costo-beneficio y versatilidad con las máquinas que tienen aplicaciones similares que son comerciales hoy en día. Por último se proponen algunas ideas para realizar trabajos futuros sobre el robot que se ha diseñado y que a consideración de los autores pueden ser las siguientes etapas para consolidar un producto más eficaz y funcional.

Palabras clave: Robot Planar, robótica paralela, cadena cinemática, cinemática directa, cinemática inversa, análisis jacobiano, Matlab, Arduino.

ABSTRACT

The present document contains in a detailed form the previous needed knowledge to understand the parallel robotic, and the way that in the history And the form in which across the history there have been realized contributions that formed what nowadays is known as robotics. In the next chapters it will focus in the design and construction of a planar parallel robot with three freedom degrees rotational, were devoted some sections of the present document to mathematician model of the system that arises from the geometrical analysis of Movement, Continues with the obtaining of the equations that allows to calculate the linking movements of each one of his kinematics chains and his final efector, the modelling stage finish with jacobian matrix analysis that allow to calculate the linking speeds and the mobile platform that includes his final efector. This document has between his epigraphs the mechanical design of the system and the control architecture implemented. In the last chapters It is located the designed prototype inside the current Colombian market, doing a Comparison between functionality, cost-benefit relation and versatility with the machines that have similar applications that are commercialized nowadays. Finally are proposed some ideas for doing future works about this robot that has been designed and that to consideration of the authors can be the next stages for to consolidate a more effective and functional product.

Key words: Planar robot, parallel robotics, kinematic chain, direct kinematics, inverse kinematics, Jacobian analysis, Matlab, Arduino.

INTRODUCCION

Con la evolución del pensamiento humano, surgió la necesidad de facilitar tareas de la vida cotidiana que de cierta forma eran tediosas y repetitivas, por esta razón la humanidad empezó a desarrollar elementos que permitieran realizar dichas labores de una forma más eficiente. Con el crecimiento demográfico exponencial que la humanidad ha tenido como rasgo principal a lo largo de su recorrido en el planeta, la mentalidad del ser humano cambió no solo en los hogares de la gente del común, sino que como un efecto dominó generó un cambio de visión en el sector industrial, impulsando la evolución tecnológica del hombre.

Debido a la necesidad de reducir costos y tiempo de producción en las industrias, la humanidad se vio obligada a crear máquinas que realizaran tareas que los humanos no podían realizar con total eficiencia. Debido al auge de la década de los 90 de la industrialización y automatización de las plantas de producción, las empresas buscaron armarse con un arsenal de robots de última tecnología con el único objetivo de maximizar las ganancias. Hoy en día la mayoría de las empresas utilizan los robots de tipo serial, sin embargo desde hace unos años ha venido cambiando esta tendencia y los mecanismos paralelos han adquirido una mayor demanda debido a la investigación, a sus características y propiedades particulares, tales como su velocidad y la capacidad de soportar cargas mucho mayores a su peso debido a su gran cantidad de puntos de apoyo. Entre las principales desventajas que se pueden destacar frente a los robots seriales se pueden enunciar: el reducido espacio de trabajo y la complejidad de sus cálculos.

En este documento se plantean los métodos y procedimientos correctos e idóneos que se deben ejecutar para diseñar un robot paralelo del tipo Planar que se mueve en el plano $[x,y]$, por medio de tres articulaciones rotativas (RRR), el cual tiene como propósito fines educativos didácticos, campo en el que este prototipo puede tener gran influencia en los jóvenes profesionales.

1. DESCRIPCION DEL SISTEMA

1.1 PLANTEAMIENTO DEL PROBLEMA

Cada generación humana ha incentivado el desarrollo de la robótica, con cada aporte se han desarrollado dispositivos que ni siquiera se habían podido imaginar en épocas anteriores y que hoy en día son una realidad gracias al desarrollo tecnológico que ha venido de la mano con la evolución humana, en este instante de la historia, la humanidad siente en mayor magnitud el auge de las nuevas tecnologías y por tal razón se han impulsado las carreras afines, hoy en día es normal que los jóvenes deseen estudiar carreras relacionadas con la robótica, sin embargo, existen muchos retos que la educación en Colombia aún no ha podido superar, entre ellos se puede encontrar la poca integración de horas de estudio prácticas en los programas profesionales, lo que genera que al culminar la carrera, el profesional recién egresado tenga vacíos en el ámbito práctico y a su vez estos vacíos limiten sus oportunidades de ingresar al mercado laboral. Los laboratorios de universidades que tienen acceso a mecanismos robóticos de alto nivel de empresas reconocidas son pocos, y casi ninguno invierte en robots de cadena cinemática cerrada debido al costo tan elevado de estos sistemas y del poco conocimiento que existe en nuestro país hacia este tipo de robótica, es importante que se puedan adquirir conocimientos tanto prácticos como teóricos de la robótica en especial de la paralela que permitan desarrollar futuros profesionales con mejores bases para enfrentar la vida laboral.

1.2 ANTECEDENTES

A lo largo de la historia humana, ha surgido la necesidad de crear máquinas que imiten los movimientos de los seres vivos, los griegos las llamaron *Automatos*, de ahí la palabra autómatas, que significa que la máquina imita movimientos de un ser¹.

El conocido *Leonardo Da Vinci* en los siglos XV y XVI construyó el *León Mecánico* el cual fue un regalo para el rey Luis XII, este consistía en mostrar el escudo con las armas, abriendo su pecho².

En 1917 se hizo una obra checoslovaca denominada Rossum's Universal Robots con este se dio nombre al robot, que significa trabajo forzado que viene de la palabra eslava que significa: *que hace los trabajos de forma forzada*³.

En los siglos XVII y XVIII se crean en Europa muñecos mecánicos con algunas características propias de los robots, como por ejemplo Jacques De

¹ BARRIENTOS, Antonio, *et al.* Fundamentos De Robótica. En: Introducción. España. 2 ed. p.2

² *Ibid.*, p. 2.

³ *Ibid.*, p. 3.

Vaucansos construyó algunos muñecos musicales (Figura 1) capaces de interpretar la flauta, Su fin era el entretenimiento, aun así este sistema fue una muestra del desarrollo en la tecnología de esa época⁴.

Figura 1. Muñeco musical creado por Jacques De Vaucansos.



Henri Maillardert en 1805 diseñó una muñeca mecánica que tenía la capacidad de hacer dibujos (Figura 2), mediante un sistema de levas. Otras creaciones mecánicas de forma humana se crearon en la revolución industrial, las cuales tenían como principal campo de acción el sector textil⁵.

Figura 2. Muñeca de levas inventada por Henri Maillardert.

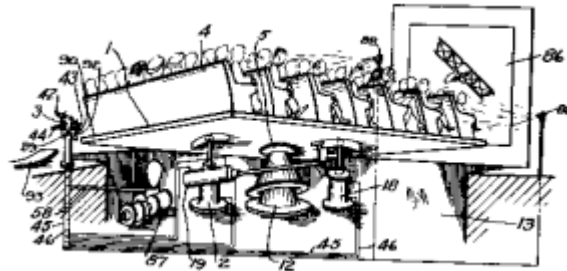


⁴ LLANOS, Fátima. Lliguin,Angel. Control Electronico por comandos de voz para un robot zoomórfico tipo mascota interactiva. En: Robótica. Ecuador. 29.p

⁵ Ibid., p. 30.

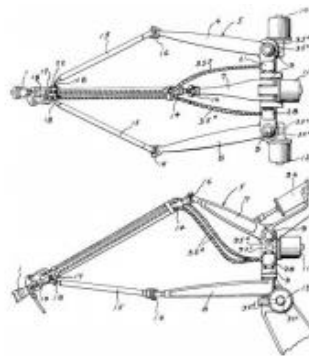
La plataforma de movimiento destinada a la industria del entretenimiento fue patentada en el año 1931, y fue la primera de su categoría (Figura 3) este nunca fue construido pero fue diseñado por James E. Gwinnett⁶.

Figura 3. Supuesto primer diseño de un robot paralelo espacial.



En 1940 se desarrolló un robot de cinco grados de libertad que fue utilizado para la labor de pintar, el robot tenía tres brazos y en cada uno poseía dos eslabones, estos estaban unidos por una misma junta (Figura 4), su orientación estaba dada por dos actuadores para la ubicación del spray⁷.

Figura 4. Primer robot paralelo.



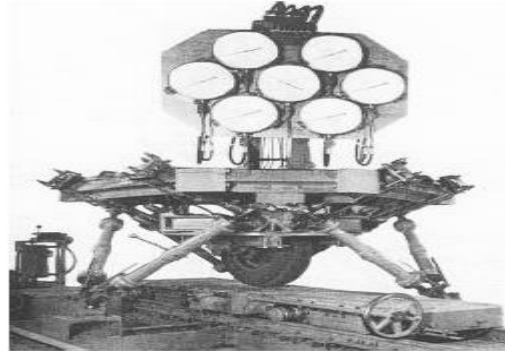
En 1974 el Doctor Eric Gough diseñó un octaedro Hexápodo, con sus eslabones variables, que servía para comprobar el comportamiento de los neumáticos (Figura 5) aplicando cargas en sus diferentes ejes, con esto se intentaba simular el aterrizaje de un avión, en su época fue la plataforma más importante y más popular, hoy en día existen muchas plataformas basadas en

⁶ ARACIL, Rafael. et.al. ROBOTS PARALELOS: Maquinas con un pasado para una robotica del futuro En: Introducción. Madrid. vol.3. no.1. p.17.

⁷ Ibid., p. 17.

este proyecto como por ejemplo el prototipo de nombre de MAST (Multi-Axis Simulation Table)⁸.

Figura 5. Plataforma de Gough.



Sobresalen también los hexápodos de seis grados de libertad, que proporcionan tres rotacionales y tres traslacionales (Figura 6), su funcionamiento se caracteriza por tres servos fijos sobre una base móvil, es capaz de mover objetos a una velocidad alta, llegando hasta cincuenta veces la gravedad⁹.

Figura 6. Robot hexápodo



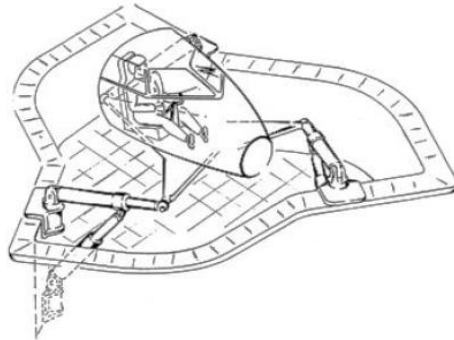
En 1965 se presentó un artículo a través de Mr. Stewart, dicho documento explicaba cómo era el funcionamiento de la plataforma de movimiento de seis grados de libertad utilizada para un simulador de vuelo(Figura 7), es muy

⁸ Ibid., p. 18.

⁹ VIVAS, Andres. robótica paralela: aplicaciones industriales, modelado y control En: Introducción. Universidad del Cauca, Departamento de Electrónica, Instrumentación y Control, Popayán, Colombia .2006.

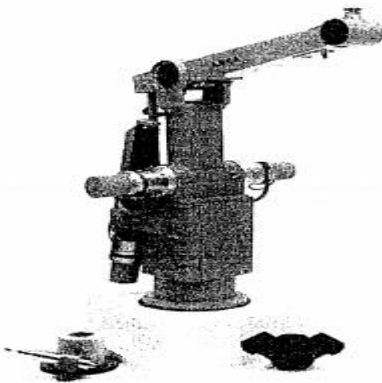
diferente al presentado anteriormente, este robot tiene gran influencia en el mundo escolar¹⁰.

Figura 7. Plataforma de Stewart 1965.



En 1973 en Europa se construyó el primer robot con accionamiento eléctrico (figura 8), el IRb6 , años más tarde se produjo el IRb60 , también en 1982 se creó en la Universidad de Yamanashi de Japón el concepto de robot SCARA, (Selective Compliance Assembly Robot Arm), que buscaba realizar ensambles de piezas con tres o cuatro grados de libertad¹¹

Figura 8. Robot IRb6.



En la década de los noventa se introdujo este tipo de robótica dentro de máquinas cortadoras laser de diferentes materiales, entre ellos principalmente los metales usados en las empresas de fabricación metalmecánica, a continuación se puede observar un claro ejemplo de una maquina cortadora que implementa la robótica de cadena cinemática cerrada con el fin de soportar grandes cargas y obtener un mayor grado de precisión:

¹⁰ ARACIL. Op. cit., p. 18.

¹¹ BARRIENTOS. Op. cit., p.7.

Figura 9. Maquina cortadora laser DURMA.



Ya en un plano más avanzado se encuentran investigaciones de diferentes estudiantes, en este trabajo se desarrolló un robot 3RRR (Figura 9), Finalmente este documento presenta el desarrollo cinemático inverso y los cálculos del espacio de trabajo, para finalizar con un análisis jacobiano inverso¹².

Figura 10. Prototipo de robot 3RRR



1.3 DESCRIPCIÓN DEL PROBLEMA

Hoy en día las carreras profesionales en Colombia dedican la mayor parte de su programa y horas de estudio a la parte teórica, sin embargo cuando los estudiantes terminan su carrera y se convierten en profesionales que deberían

¹² MARTINEZ, Noe. ROJAS, Angel reproducción de trayectorias en un robot paralelo 3rrr configurable. Coyoacán, México, D.F En: Consideraciones al manipulador.p.446.

estar preparados para enfrentar el mundo laboral, encuentran una barrera que con el paso del tiempo se denomina la experiencia laboral, es decir que a consideración de los empresarios un profesional que sea recién egresado no puede aspirar a un cargo profesional debido a la falta de experiencia, razón por la cual la mayoría de ofertas laborales para cargos profesionales requieren una determinada cantidad de años de experiencia, limitando o excluyendo de manera inmediata a los jóvenes profesionales. Este inconveniente se puede evidenciar en todos los sectores profesionales de Colombia, sin embargo este fenómeno se presenta con mayor frecuencia en el sector industrial, es decir que los principales afectados por la problemática social del país son los profesionales recién egresados de carreras relacionadas con la ingeniería, es imperativo afirmar que uno de los campos más afectados por esto, es el campo relacionado con la automatización, en ella principalmente los jóvenes ingenieros mecatrónicos que se encuentran limitados por el poco desarrollo tecnológico que el país puede ofrecer, debido a la poca inversión en investigación y desarrollo por parte del gobierno nacional. Adicionalmente como este tipo de carreras son relativamente nuevas en el país si se compara con otras, la mayoría de universidades no poseen la infraestructura y maquinaria de laboratorio necesaria para solventar una completa educación práctica de los profesionales, y el enfoque que normalmente puede encontrarse en los laboratorios de universidades que llevan más tiempo ofreciendo programas afines, normalmente se orientan en la robótica de tipo serial excluyendo la robótica paralela que en las grandes economías industrializadas tiene una gran acogida, limitando de manera directa el desarrollo práctico del estudiante.

1.4 FORMULACIÓN DEL PROBLEMA.

¿Cómo podrían las futuras generaciones de ingenieros que estudien carreras afines con la robótica vincularse desde su pregrado con los robots paralelos y como se puede aportar a un buen desarrollo teórico-práctico de esta tecnología en la futura industria colombiana?

1.5 JUSTIFICACIÓN

Es importante proporcionarles a los jóvenes profesionales una educación no solo teórica si no práctica que les garantice una amplia gama de oportunidades dentro del campo laboral cuando logren culminar sus estudios, el presente proyecto busca diseñar un prototipo que logre suplir los vacíos prácticos que hoy en día los jóvenes estudiantes de carreras afines con la robótica pueden presentar, de tal manera que este prototipo se ubique dentro de la robótica paralela, con el fin de proporcionarles a los estudiantes una visión integral del campo de la robótica en general, permitiéndoles conocer las diferentes aplicaciones, ventajas y desventajas que este tipo de mecanismos puede ofrecer, determinando cuales pueden ser las funciones de este tipo de robots que pueden ser implementadas dentro del ámbito laboral cuando sean

profesionales. El proyecto se enfocará en el ámbito didáctico y educativo, tendrá como principales beneficiarios a todos los futuros estudiantes de carreras afines con la robótica. Es un proyecto innovador que busca el desarrollo complementario de la robótica de forma más práctica en los futuros ingenieros y permitirá que se creen espacios en las universidades que estarán dedicados al estudio de todos los aspectos de la robótica paralela. Principalmente el proyecto estará encaminado a realizar un software en el que el usuario sea capaz de generar diferentes configuraciones del robot prototipo, que puedan modificar de forma virtual todos los aspectos o variables usadas en el cálculo del robot para que puedan realizar las conclusiones respectivas frente a dichos cambios, puedan generar informes que incluyan gráficos y puedan ser analizados más fácilmente por los estudiantes.

1.6 OBJETIVO GENERAL

Diseñar y construir un robot paralelo Planar de 3 grados de libertad, dados por 3 servomotores que generan articulaciones rotacionales. Este prototipo debe suplir necesidades didácticas e industriales dependiendo de la aplicación que se le dé y debe ser capaz de llegar a un punto arbitrario escogido por el usuario dentro de su área de trabajo.

1.7 OBJETIVOS ESPECÍFICOS

- Resolver las ecuaciones de cinemática directa e inversa.
- Comparar distintos espacios de trabajo del prototipo con el fin de que los estudiantes puedan conocer ventajas y desventajas de los robots paralelos planares.
- Desarrollar el modelamiento matemático para el control del movimiento del robot
- Realizar el diseño mecánico del prototipo.
- Calcular la posición y orientación de la plataforma móvil con base a datos provistos por el usuario a través de matlab.
- Obtener un grado de precisión mayor o igual al 85%
- Conocer las aplicaciones y limitaciones del prototipo
- Determinar la diferencia de costos en fabricación del prototipo y producción, comparándolos con robots existentes en el mercado.

1.8 ALCANCES Y LIMITACIONES

El proyecto de grado puede desarrollarse en diferentes campos, principalmente en el campo de la pedagogía en el cual se puede adaptar para que los estudiantes de carreras relacionadas con la robótica, puedan entender de forma más eficiente y directa el universo que envuelve a la robótica paralela. Dentro del campo de la investigación se creó una interfaz gráfica en la cual los estudiantes pueden observar en tiempo real el comportamiento de cada uno de los motores, visualizando el Angulo que forman respecto al plano con el que se analice el movimiento y las coordenadas de la plataforma móvil en función de los ángulos formados por las articulaciones. Ya que se fabricó un robot cuya cadena cinemática no es común debido a que es cerrada, los usuarios pueden adquirir un conocimiento teórico-práctico de la robótica paralela obteniendo sus propias conclusiones y realizando prácticas en las que podrán interactuar en tiempo real con el mecanismo, de igual forma se puede analizar el espacio de trabajo del robot en forma de simulación con el fin de verificar el máximo tamaño en el que el robot puede seguir las coordenadas que el usuario desee que se introduzca por medio de la interfaz gráfica de MATLAB. Las limitaciones del proyecto son el poco espacio de trabajo en comparación el tamaño del robot (2:1) debido a la restricción de los grados de libertad, el error generado en la finalización del dibujo replicado debido a la suma de los errores previos (Grado de precisión) en cada punto que el robot tome como coordenada para definir su trayectoria debido a la resistencia física que otorgan las juntas pasivas del prototipo y el tiempo que tarda en replicar la imagen deseada que es directamente proporcional a la cantidad de puntos requeridos para copiar la figura introducida por el operario.

2. ROBÓTICA

La definición de robot varía según la fuente consultada, según Juan Antonio Briones¹³, un robot es una maquina programable para realizar tareas, este lleva a cabo instrucciones y se creó para realizar trabajos que el ser humano no puede realizar fácilmente.

2.1 ROBOTS INDUSTRIALES

La definición de un robot industrial puede variar, ya que Europa junto con América, y Japón tienen distintas definiciones, en los primeros según la *Asociación de industrias de robóticas (RIA)*, un robot industrial es una maquina multifuncional reprogramable, con capacidad de cambiar piezas, herramientas o cualquier tipo de dispositivo, dependiendo las trayectorias, se podrán hacer distintas tareas, para los japoneses es una unidad mecánica que posee articulaciones mecánicas capaces de manipular¹⁴.

2.1.1 Tipos y clasificación de robots industriales Existen cuatro tipos de robots industriales según la Federación internacional de robótica (IFR)¹⁵:

- Secuencial
- Trayectoria controlable
- Adaptivo
- Telemanipulado

Están clasificados en generaciones¹⁶ según sus características y su función:

- Primera generación: Repite la tarea programada secuencialmente, no toma en cuenta alteraciones de su entorno.
- Segunda generación: Toma alguna información del entorno y respecto a esta, actúa conforme a sus necesidades.
- Tercera generación: La programación se hace mediante un lenguaje.

¹³ Microsoft Corporation. "Encarta Enciclopedia 2008". Citado por BRIONES, Juan. Diseño, análisis y construcción de un robot paralelo traslacional. Queretaro, 2009. p.5.

¹⁴ BARRIENTOS. Op. cit., p.9.

¹⁵ Ibid., p. 11.

¹⁶ Ibid., p. 11.

2.2 ROBOTS TELEOPERADOS Y DE SERVICIO

Los robots teleoperados son aquellos que son controlados a la distancia por una persona desde un punto fijo o estación, su ventaja es que para el usuario no existe ningún tipo de riesgo, ya sea por manipular materiales peligrosos como químicos y sustancias peligrosas, estos robots son muy utilizados por las fuerzas armadas (Figura 11) para el control de explosivos¹⁷.

Figura 11. Estación de mando con su robot teleoperado.



Los robots de servicio según la *federación internacional de robótica* son aquellos que prestan servicios útiles a los seres humanos, estos operan de manera semi y automática¹⁸, sin contar operaciones de fabricación.

2.3 CLASIFICACIÓN DE ROBOTS

Dependiendo la estructura cinemática como lo es de lazo abierto y de lazo cerrado se pueden catalogar los distintos tipos de robots como nos dice *Briones*¹⁹.

- Robots seriales.
- Robots híbridos.
- Robots paralelos.

Para que un robot sea serial su cinemática tiene que estar en una cadena de lazo abierto, también se puede afirmar que puede ser un robot híbrido siempre y cuando posea los dos tipos de cadenas cinemáticas como lo son cerradas y

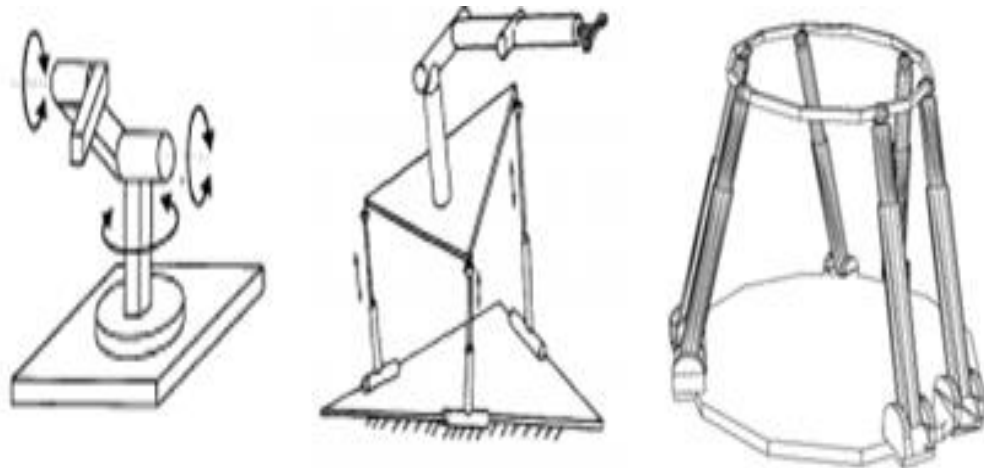
¹⁷ CORREA, Alexander. Sistemas robóticos teleoperados. En: Ciencia e ingeniería neogranadina. noviembre, 2005. no. 15, p. 63.

¹⁸ Federación internacional de robótica. Citado por Aracil,Rafael.ARMADA,Manuel.BALAGUER,Carlos, Robots de servicio. Madrid.2008.p. 6.

¹⁹ BRIONES, Juan. Diseño, análisis y construcción de un robot paralelo traslacional.Queretaro.En: Ciencia de investigación en ciencia aplicada y tecnología avanzada .queretaro. 2009. p.5.

abiertas, por lo tanto provee características de los robots paralelos y en serie²⁰ en la (figura 12) se darán a conocer los distintos tipos de robots.

Figura 12. Robot seriar, híbrido y paralelo



Por último tenemos el robot paralelo, que es aquel que tiene su plataforma móvil conectada a una base, por medio de cadenas cinemáticas independientes como dice *Briones*²¹.

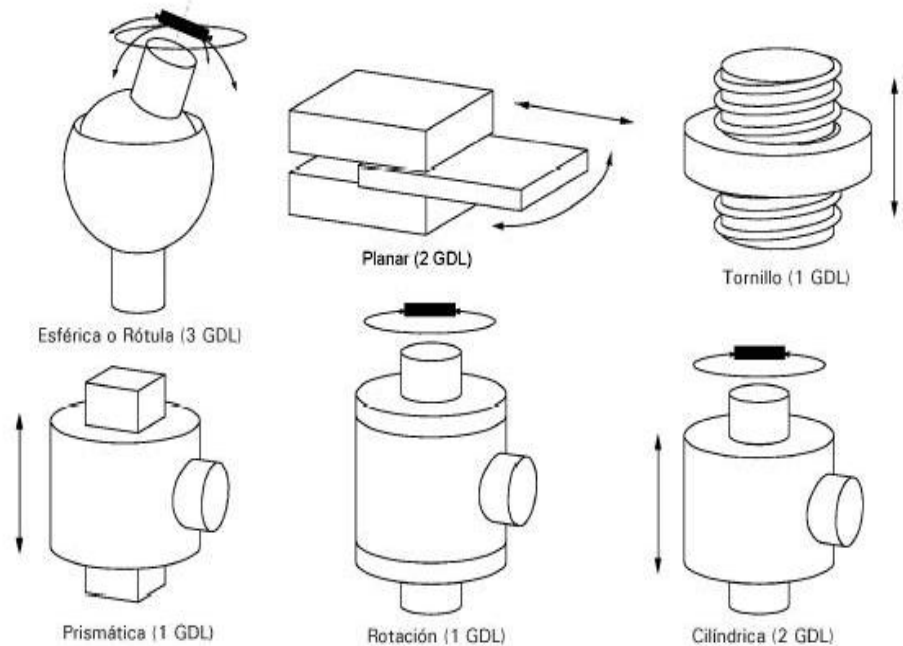
2.3.1. Estructura mecánica del robot Gran parte de los robots industrial es se asemejan al ser humano, más exactamente a sus brazos, hablando de la parte mecánica, se encuentra formada por eslabones y ellos a su vez están conectados mediante articulaciones, las cuales permiten el movimiento del brazo, existen diferentes tipos de articulaciones como lo son de giro y desplazamiento, logrando así la unión de estos dos y generando diferentes tipos de articulación (Figura 13)²², las más utilizadas son las prismáticas y las de rotación.

²⁰ Tsai L.W, *Robot Analysis. The mechanics of serial and parallel manipulators*. EUA: Wiley Interscience, 1999. Citado por BRIONES, Juan. Diseño, análisis y construcción de un robot paralelo traslacional. Queretaro. queretaro. 2009. p.5

²¹ Ibid., p. 5.

²² Gil, Jorca. Robótica e inteligencia artificial, Componentes/elementos de un sistema robot [en línea] <http://www.pagines.fib.upc.es/~rob/protegit/treballs/Q2_03-04/rob_ia2/elem1.htm> [citado en 15 de septiembre de 2015]

Figura 13. Tipos de articulaciones



2.4 ROBOTS PARALELOS

Como se dijo en los puntos anteriores los robots paralelos se pueden definir como un robot que posee una unión de varias cadenas cinemáticas independientes, estos son utilizados en su mayoría para suplir algunas deficiencias que tienen los robots seriales, algunas de las ventajas de este robot son²³:

- La relación entre carga y tamaño es muy elevada, debido a que sus eslabones se conectan a la base móvil del robot, lo que permite que todos los elementos se muevan de forma simultánea y proporcionen una mayor cantidad de puntos de apoyo, de tal forma que se requiere una carga muy alta para generar falla en robots de este tipo, a diferencia de los robot de tipo serial²⁴.
- Tienen gran precisión por su rigidez²⁵

Dependiendo del tipo de aplicación se puede decir que tienen algunas desventajas:

²³ ARACIL. Op. cit., p.18

²⁴ Ibid., p. 19

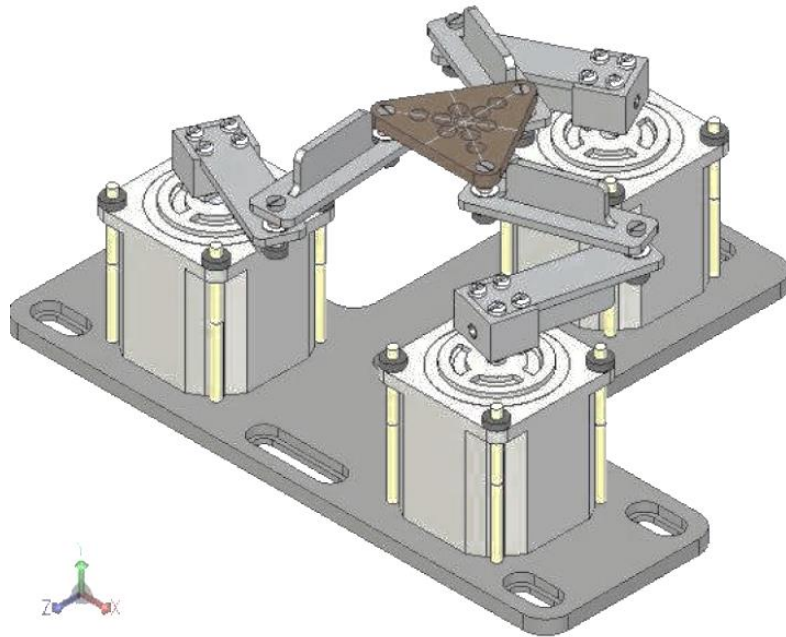
²⁵ Ibid., p. 19

- El espacio de trabajo de trabajo es reducido a comparación de otros tipos de robots²⁶.
- El cálculo de la cinemática es más complicada²⁷

2.4.1 Configuraciones de los robots paralelos Los robots paralelos se clasifican en función del tipo de eslabones, articulaciones y cadenas cinemáticas. También se pueden clasificar por el tipo de movimiento, en los que se pueden encontrar los robots planares, esféricos y espaciales²⁸.

2.4.1.1 Robots paralelos planares Los robots paralelos solo pueden tener 2 o 3 grados de libertad un ejemplo es (figura 14), por lo tanto su movimiento solo puede ser a un solo plano²⁹

Figura 14. Robot planar



Los grados de libertad para un robot paralelo Planar se pueden calcular mediante el criterio de criterio de Grübler-Kutzbach ec (3.1)³⁰:

²⁶ Ibid., p. 19

²⁷ Ibid., p. 19

²⁸ Tsai L.W, *Robot Analisis. The mechanics of serial and parallel manipulators*. EUA: Wiley Interscience, 1999. Citado por YAÑEZ,Ricardo. Resolución de mecanismo paralelo Planar 3RRR impulsado por actuadores eléctricos.Queretaro. noviembre.2007.p.2-3.

²⁹ Ibid., p.3.

³⁰ Ibid., p.19-20.

$$F = \lambda(n - j - 1) + \sum_i f_i \quad (3.1)$$

Donde

F = Grados de libertad del mecanismo.

λ = Grados de libertad en el espacio el cual se desea trabajar.

n = Numero de eslabones con base fija incluida.

j = Numero de juntas cinemáticas.

f = Grados de libertad permitidos por las juntas i .

Aplicando la ecuación 3.1 para el caso del prototipo planar que se diseñó y fabricó:

$$\lambda = 3;$$

$$n = 8;$$

$$j = 9;$$

$$F = 3 - (8 - 9 - 1) + 9 \cdot 1 = 3$$

En este caso nuestro robot tiene 3 Grados de libertad, estas se identifican por poseer sus tres cadenas cinemáticas que van vinculadas con una junta pasiva y otras dos activas, existen 21 formas de hacer un robot paralelo Planar con las juntas prismáticas y de revolución (tabla 1)³¹.

Tabla 1. Distintas combinaciones de juntas cinemáticas

<u>RRR</u>	<u>RPR</u>	<u>RPP</u> ∞	<u>PRR</u>	<u>PRP</u>	<u>PPR</u>	<u>RRP</u>
<u>RRR</u>	<u>RPR</u>	<u>RPP</u>	<u>PRR</u>	<u>PRP</u> ∞	<u>PPR</u> \approx	<u>RRP</u> \approx
<u>RRR</u> \approx	<u>RPR</u> \approx	<u>RPP</u> \approx	<u>PRR</u> \approx	<u>PRP</u> \approx	<u>PPR</u> ∞	<u>RRP</u> \approx

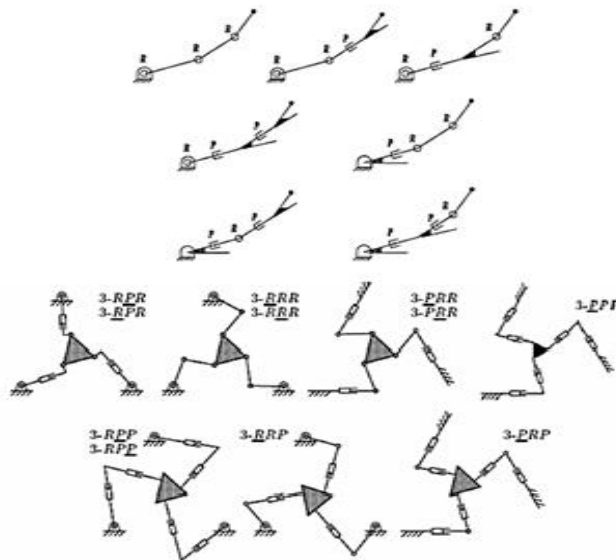
Según el autor de Resolución del mecanismo paralelo Planar 3RRR impulsado por actuadores eléctricos³², tres de estas combinaciones marcadas con el símbolo infinito solo dan un grado de libertad y ocho marcadas con el símbolo aproximado están con mecanismos paralelos iguales, y al acabar con las

³¹ Ibid., p.19-20.

³² YÁÑEZ, Ricardo. Resolución de mecanismo paralelo Planar 3RRR impulsado por actuadores eléctricos. En: centro de investigación en ciencia aplicada. Queretaro. noviembre.2007.p.2-3.

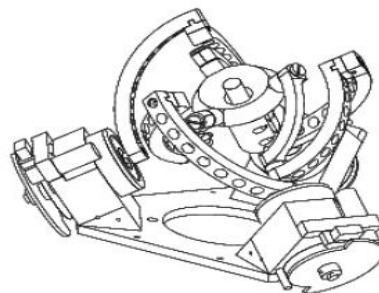
repeticiones ya que no importa si la junta es pasiva o activa, quedan siete combinaciones en serie (figura 15), como también se pueden usar tres cadenas iguales.

Figura 15. Siete combinaciones en serie con su respectiva cadena cinemática cerrada³³



2.4.1.2 Robots paralelos esféricos En los mecanismos paralelos (Figura 16) esféricos su movimiento tiende a formar curvas sobre un punto determinado, formando una esfera.³⁴

Figura 16. *The agile eye*, mecanismo esférico.

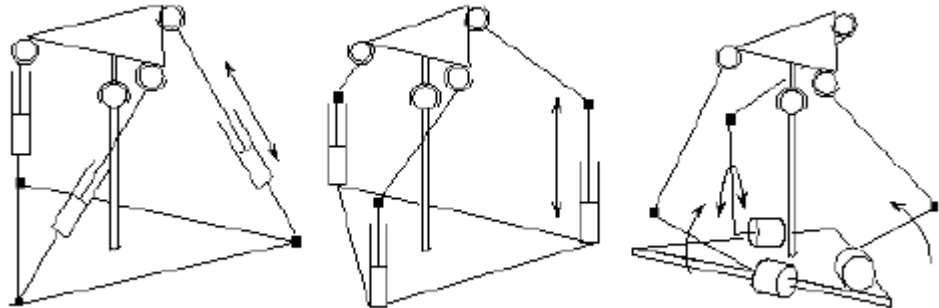


³³ Williams II R. L. and Shelley B. H. Inverse Kinematics for planar parallel manipulators. American Society Of Mechanical Engineers, 1997 citado por YAÑEZ, Ricardo. Resolución de mecanismo paralelo... Queretano. noviembre 2007. p.21

³⁴ YAÑEZ, Ricardo. Resolución de mecanismo paralelo Planar 3RRR impulsado por actuadores eléctricos. Queretaro. noviembre. 2007. p.3.

2.4.1.3 Robots paralelos espaciales Los robots paralelos espaciales (Figura 17) se caracterizan por tener la capacidad de desenvolverse en todo el espacio tridimensional, pueden tener hasta 6 grados de libertad³⁵.

Figura 17. Robots paralelos espaciales.

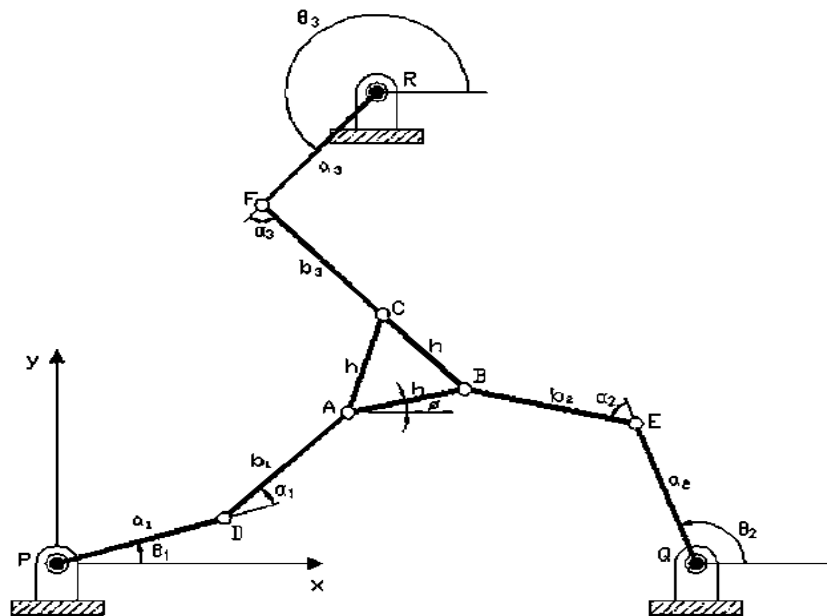


³⁵ Ibid., p.4.

3. ROBOT PLANAR PARALELO 3RRR

Gracias a que en el capítulo anterior se dieron a conocer los diferentes tipos de robots, que existen, el presente capítulo se destina a describir en profundidad el *Robot Planar paralelo 3RRR* (Figura 18), este consta de tres servomotores anclados a una base fija, entre los tres se forma un triángulo equilátero, cada uno de estos está acoplado a un eslabón a_i , el cual tiene la capacidad de albergar otro eslabón b_i por medio de una junta rotacional, se tienen tres puntas o extremos en los cuales se acopló un triángulo equilátero de lado h y que es la plataforma móvil del robot, en el que se acopla el instrumento para realizar el dibujo, para este proyecto es un lápiz, que está ubicado en el centro de la plataforma móvil.

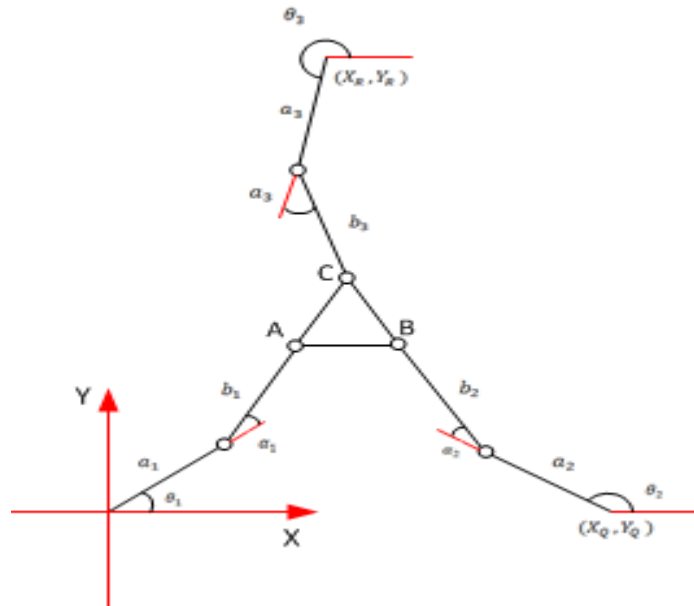
Figura 18. Robot paralelo planar, 3RRR



3.1 ANÁLISIS GEOMÉTRICO

Como podemos observar en (Figura 19), El origen del plano coincide con el primer punto coordinado P de nuestro robot, Como la distancia entre servomotores es igual ya que forman un triángulo equilátero podemos decir que $\overline{PQ} = \overline{QR} = \overline{RP}$ lo mismo pasa con el efector final $\overline{AB} = \overline{BC} = \overline{CD}$.

Figura 19. Manipulador paralelo 3RRR



Después de realizar una suma de vectores y por cada cadena cinemática hallar las componentes en $(X$ y $Y)$, se reemplaza una ecuación en términos de otra, logramos obtener tres ecuaciones en términos de $[X_A, X_A\theta_i]$, donde $i = 1, 2, 3$ ³⁶. El desarrollo de estas tres ecuaciones (4.0 4.1 y 4.2) se obtiene en el apéndice A.

Cadena cinemática 1.

$$X_A^2 + Y_A^2 - 2X_A a_1 \cos(\theta_1) - 2Y_A a_1 \sin(\theta_1) + a_1^2 - b_1^2 = 0 \quad (4.0)$$

Cadena cinemática 2.

$$X_A^2 + Y_A^2 - 2X_A X_Q - 2Y_A Y_Q + X_Q^2 + Y_Q^2 + h^2 + a_2^2 - b_2^2 + 2X_A h \cos(\phi) + 2Y_A h \sin(\phi) - 2X_A a_2 \cos(\theta_2) - 2Y_A a_2 \sin(\theta_2) - 2h a_2 \cos(\phi) \cos(\theta_2) - 2h a_2 \sin(\phi) \sin(\theta_2) + 2a_2 X_Q \cos(\theta_2) + 2a_2 Y_Q \sin(\theta_2) - 2h X_Q \cos(\phi) - 2h Y_Q \sin(\phi) = 0 \quad (4.1)$$

³⁶ Tsai L.W., *Robot Analysis. The mechanics of serial and parallel manipulators*. EUA: Wiley Interscience, 1999. Citado por YAÑEZ, Ricardo. Resolución de mecanismo paralelo Planar 3RRR impulsado por actuadores eléctricos. Queretaro. noviembre. 2007. p.23

Cadena cinemática 3.

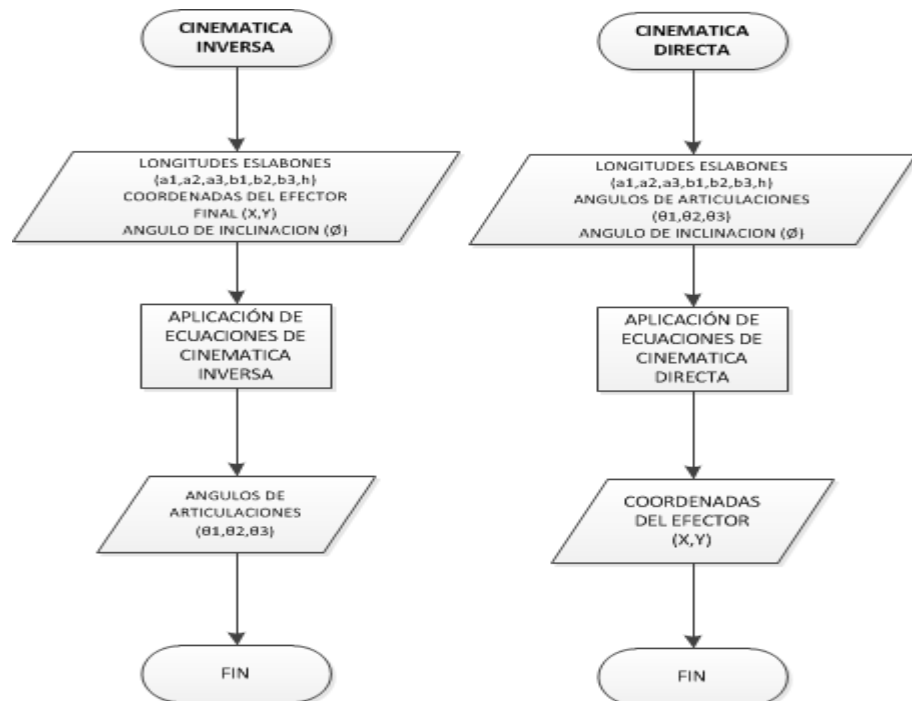
$$\begin{aligned}
 X_A^2 + Y_A^2 - 2X_A X_R - 2Y_A Y_R + X_R^2 + Y_R^2 + h^2 + a_3^2 - b_3^2 + 2X_A h \cos\left(\phi + \frac{\pi}{3}\right) + 2Y_A h \sin\left(\phi + \frac{\pi}{3}\right) - 2X_A a_3 \cos(\theta_3) - 2Y_A a_3 \sin(\theta_3) - 2h a_3 \cos\left(\phi + \frac{\pi}{3}\right) \cos(\theta_3) - 2h a_3 \sin\left(\phi + \frac{\pi}{3}\right) \sin(\theta_3) + 2a_3 X_R \cos(\theta_3) + 2a_3 Y_R \sin(\theta_3) - 2h X_R \cos\left(\phi + \frac{\pi}{3}\right) - 2h Y_R \sin\left(\phi + \frac{\pi}{3}\right) = 0
 \end{aligned} \tag{4.2}$$

Estas ecuaciones son útiles para desarrollar los cálculos siguientes de las cinemáticas, que establecen una relación entre los valores del punto A y la plataforma móvil.

3.2 ANÁLISIS DE POSICIÓN.

La cinemática del robot estudia el movimiento del mismo respecto a un sistema de referencia, su relación entre la orientación del efector final y la posición, dependiendo de los valores de las coordenadas articulares, existen dos problemas de cinemáticas (figura 20).

Figura 20. Cinemática directa en inversa



3.3 CINEMÁTICA INVERSA.

Esta intenta a partir de $[X, Y, \phi]$ conocidos, encontrar los ángulos $[\theta_1, \theta_2, \theta_3]$, para el primer ángulo se toma la ecuación (4.0) que se halló anteriormente y se factorizan los términos que tienen $\sin \theta_i$ y $\cos \theta_i$ (4.3)³⁷.

$$e_1 \text{Sen}(\theta_1) + e_2 \text{Cos}(\theta_1) + e_3 = 0 \quad (4.3)$$

Dónde:

$$e_1 = -2Y_A a_1$$

$$e_2 = -2X_A a_1$$

$$e_3 = X_A^2 + Y_A^2 + a_1^2 - b_1^2$$

Conociendo estas identidades trigonométricas (4.4).

$$\text{Sen}(\theta_1) = \frac{2t_1}{1+t_1^2} \quad \text{Cos}(\theta_1) = \frac{1-t_1^2}{1+t_1^2} \quad t_1 = \text{Tan}\left(\frac{\theta_1}{2}\right) \quad (4.4)$$

Sustituyendo estas identidades en la ecuación (4.3) resulta en una nueva ecuación (4.5).

$$e_1 \frac{2t_1}{1+t_1^2} + e_2 \frac{1-t_1^2}{1+t_1^2} + e_3 = 0 \quad (4.5)$$

Se simplifica la ecuación y de dicha operación se obtiene el primer ángulo (4.6) a partir de una ecuación cuadrática.

$$\theta_1 = 2\text{Tan}^{-1} \left[\frac{-e_1 \pm \sqrt{(e_1^2 - e_3^2 + e_2^2)}}{-e_2 + e_3} \right] \quad (4.6)$$

En el apéndice B se evidencia el desarrollo completo de todos los ángulos. Ya que se tratan de ecuaciones cuadráticas, se obtendrán por defecto dos posibles soluciones por cada cadena cinemática

³⁷ Tsai L.W., *Robot Analysis. The mechanics of serial and parallel manipulators*. EUA: Wiley Interscience, 1999. Citado por YAÑEZ, Ricardo. Resolución de mecanismo paralelo Planar 3RRR impulsado por actuadores eléctricos. Queretaro. noviembre.2007.p.24

3.4 CINEMÁTICA DIRECTA

A partir de $[\theta_1, \theta_2, \theta_3, a_i, b_i]$, se obtienen los valores del efector final $[X, Y, \emptyset]$.

Se escriben las ecuaciones (4.0, 4.1, 4.2) en la forma de ecuaciones según Tsai³⁸.

$$X_A^2 + Y_A^2 + e_{11}X_A + e_{12}Y_A + e_{13} = 0 \quad (4.7)$$

$$X_A^2 + Y_A^2 + e_{21}X_A + e_{22}Y_A + e_{23} = 0 \quad (4.8)$$

$$X_A^2 + Y_A^2 + e_{31}X_A + e_{32}Y_A + e_{33} = 0 \quad (4.9)$$

Resolviendo X_A, Y_A (4.10) con $\text{sen}(\emptyset), \text{cos}(\emptyset)$:

$$\delta_1^2 + \delta_2^2 + e_{11}\delta\delta_1 + e_{12}\delta\delta_2 + e_{13}\delta^2 = 0 \quad (4.10)$$

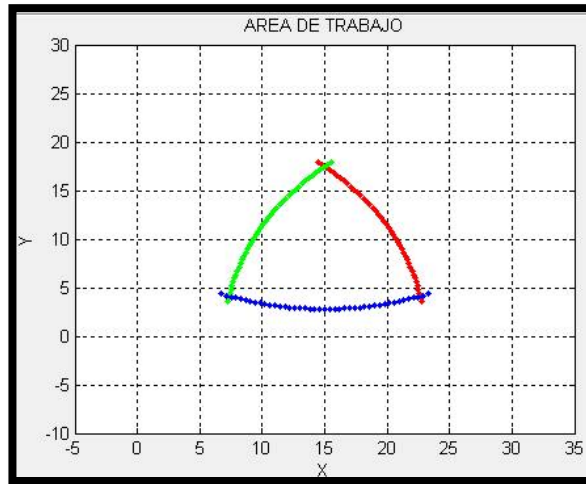
El desarrollo completo se presenta en el apéndice C. Sustituyendo la ecuación (4.4) en (4.10), se obtiene un polinomio de octavo grado, que permitirá obtener las ocho posibles configuraciones del prototipo.

3.5 ESPACIO DE TRABAJO

Para poder enviarle al robot el punto que debe alcanzar, se tiene que establecer un área donde el efector final pueda llegar sin ningún problema, teniendo en cuenta todas cadenas cinemáticas, conociendo la teoría de los robots paralelos enunciada anteriormente, se conoce que estos tienen un pequeño espacio de trabajo. El primer paso que se debe ejecutar para calcular el espacio de trabajo (figura 21) es determinar las ecuaciones que relacionen todas juntas cinemáticas con el efector final, conocer la longitud de los eslabones (a_i, b_i) , posteriormente graficar y variar el ángulo theta (θ_i) , el código se presenta en el apéndice E.

³⁸ Tsai L.W., *Robot Analysis. The mechanics of serial and parallel manipulators*. EUA: Wiley Interscience, 1999. Citado por YAÑEZ, Ricardo. Resolución de mecanismo paralelo Planar 3RRR impulsado por actuadores eléctricos. Queretaro. noviembre. 2007. p.25

Figura 21. Espacio de trabajo robot 3RRR



Como bien se mencionó en el capítulo (4.2.1) para hallar los ángulos se debe desarrollar una ecuación cuadrática, por ejemplo para el ángulo 1 en la ecuación (4.6) se obtienen dos resultados, los cuales quieren decir que existen dos configuraciones para la primera cadena cinemática, en la (figura 22,23,24, 25) se graficaron las ocho posibles configuraciones que existen para el robot Planar 3RRR, teniendo en cuenta que la longitud de los eslabones siempre es la misma y que el efector final forma un triángulo equilátero al igual que la distancia entre servos. Las juntas pasivas son aquellas que están conectadas a la plataforma móvil, y las activas las que van conectadas a los servos, es decir que las juntas activas son aquellas sobre las que se tiene control a diferencia de las pasivas. Para facilitar la comprensión se asignan nombres a la configuración de cada junta:

- Abierta (A), cuando los eslabones forman un mayor que ($>$).
- Cerrada (C), cuando los eslabones forman un menor que ($<$).

Figura 22. Configuración (CAC) y (CAA)

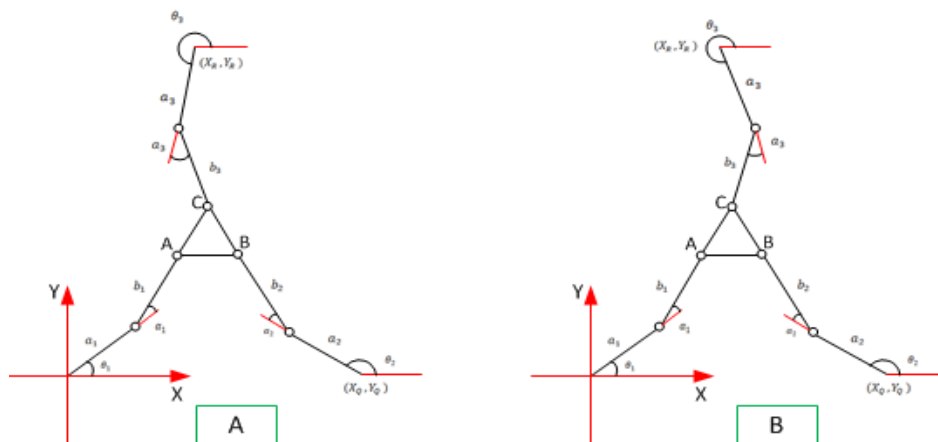


Figura 23. Configuración (CCC) y (CCA)

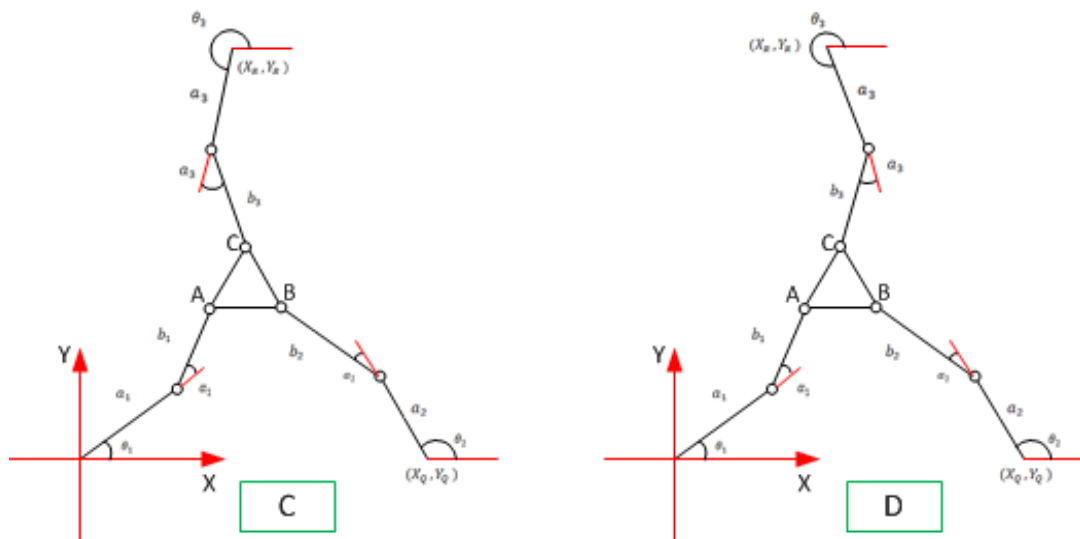


Figura 24. Configuración (ACA) y (ACC)

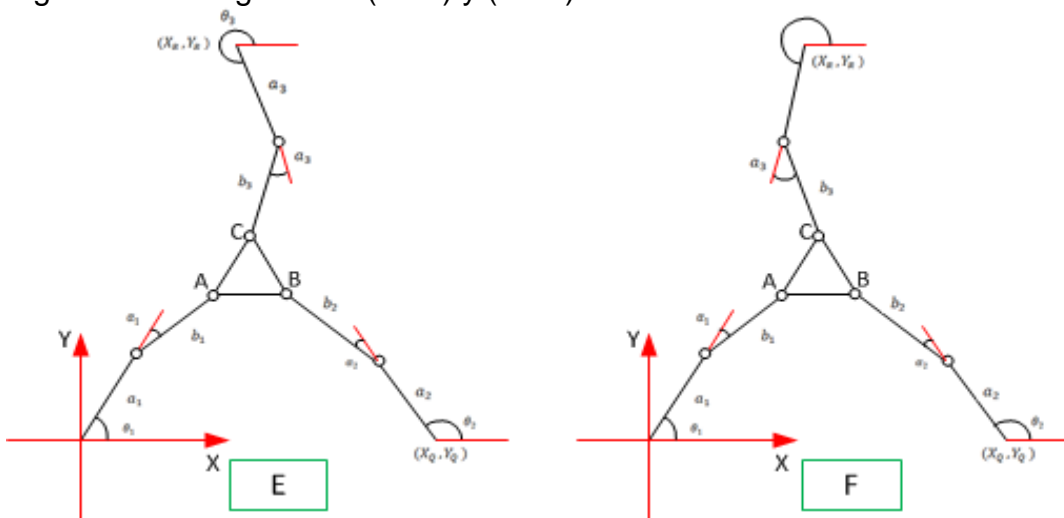
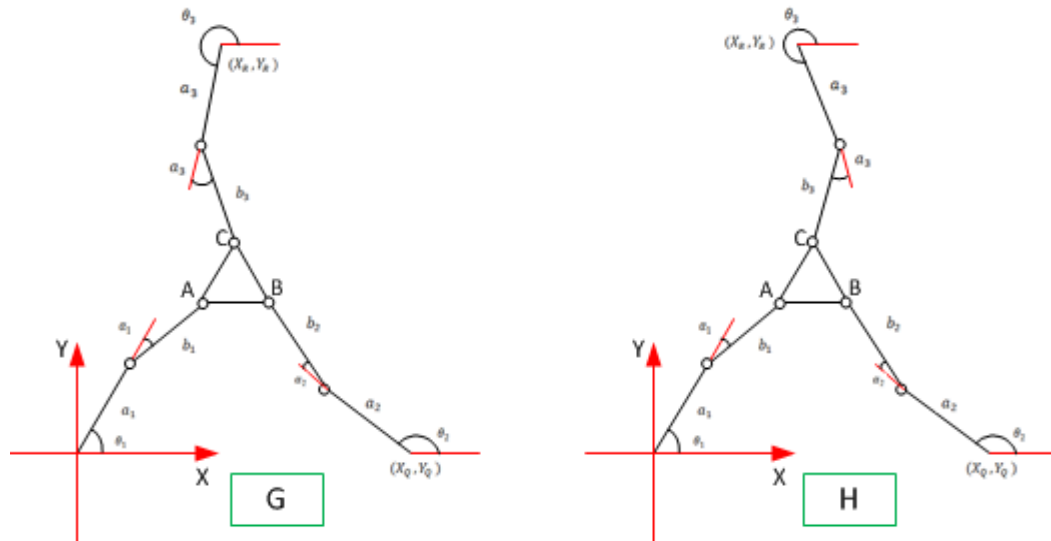


Figura 25. Configuración (AAC) y (AAA)



3.6 ANÁLISIS JACOBIANO.

El análisis jacobiano en un robot de cadena cinemática cerrada es mucho más difícil de hallar en comparación con uno del tipo serial debido a la cantidad de eslabones que forman una gran cantidad de ciclos cerrados. Este análisis es parte importante de la robótica ya que permite determinar la velocidad necesaria requerida por las articulaciones que se debe imprimir para llegar a la ubicación deseada.

Existen dos tipos de matrices jacobianas: La matriz jacobiana directa la cual permite conocer la velocidad del eslabón acoplador en función de los valores de velocidad de cada articulación y la matriz jacobiana inversa que permite obtener la velocidad de una articulación en función de la velocidad del eslabón acoplador, en el apéndice D se puede obtener el desarrollo de este análisis.

Para este prototipo la ecuación (4.11)³⁹ jacobiana describe su comportamiento, esta ecuación se traduce en la nueva ecuación (4.12)⁴⁰.

$$= \begin{bmatrix} a_{1X}b_{1Y} - a_{1Y}b_{1X} & 0 & 0 \\ 0 & a_{2X}b_{2Y} - a_{2Y}b_{2X} & 0 \\ 0 & 0 & a_{3X}b_{3Y} - a_{3Y}b_{3X} \end{bmatrix} \begin{bmatrix} V_{GX} \\ V_{GY} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}$$

³⁹ Tsai L.W, *Robot Analysis. The mechanics of serial and parallel manipulators*. EUA: Wiley Interscience, 1999. Citado por YAÑEZ, Ricardo. Resolución de mecanismo paralelo Planar 3RRR impulsado por actuadores eléctricos. Queretaro. noviembre.2007.p.25

⁴⁰ Ibid., p. 25

(4.11)

$$J_x \dot{x} = J_q \dot{q} \quad (4.12)$$

Como:

$$J = J_q^{-1} J_x \rightarrow \dot{q} = J \dot{x} \quad (4.13)$$

De tal forma que se conocen las velocidades de cada articulación a partir de una velocidad del eslabón acoplador usando la ecuación (4.13)⁴¹.

Finalmente, se puede conocer la velocidad del eslabón acoplador (4.14)⁴², a partir de las velocidades de las articulaciones con la siguiente ecuación:

$$J \dot{q} = \dot{x} \quad (4.14)$$

3.7 SINGULARIDADES

Una de las principales desventajas⁴³ que tiene los robots de cadena cinemática cerrada con respecto a los de cadena cinemática abierta es precisamente la existencia de las singularidades, en estas configuraciones el mecanismo tiende a perder rigidez mientras gana grados de libertad extra, y representa el punto en el que la estructura puede colapsar.

Las singularidades se pueden presentar cuando J_x o J_q son singulares o cuando ambas lo son, es decir cuando su determinante es igual a 0

Existen tres tipos de singularidades:

3.7.1 Singularidad de cinemática inversa Este tipo de singularidades ocurren cuando:

$$\det(J_q) = 0 \quad (4.15)$$

En esta configuración (4.15) el manipulador pierde uno o más grados de libertad, en este tipo de singularidades un robot paralelo puede resistir fuerzas o momentos en algunas direcciones. En este tipo de configuraciones el manipulador se comporta como si fuese del tipo serial

⁴¹ Ibid., p. 25

⁴² Ibid., p. 19

⁴³ Tsai L.W, *Robot Analysis. The mechanics of serial and parallel manipulators*. EUA: Wiley Interscience, 1999.

3.7.2 Singularidad de cinemática directa Este tipo de singularidades ocurren cuando:

$$\det(J_x) = 0 \quad (4.16)$$

En este tipo de singularidad (4.16) el movimiento del robot se encuentra limitado mientras todos sus actuadores están completamente bloqueados. En este caso el prototipo gana 1 o más grados de libertad, en esta configuración el robot no puede resistir fuerzas o momentos en alguna dirección

3.7.3. Singularidades combinadas Una singularidad combinada ocurre cuando los determinantes de J_x y J_q son ambos cero. Este tipo de singularidades solo pueden ocurrir en manipuladores con una arquitectura cinemática especial.

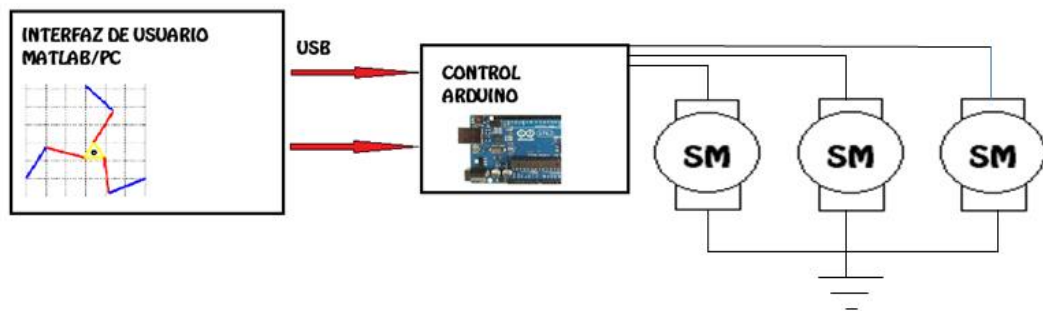
4 DISEÑO MECATRÓNICO

Con el fin de diseñar un robot paralelo se requieren varias etapas, estas son muy importantes ya que si falta una de estas, simplemente el proyecto no se puede desarrollar en la (Figura 26) detalla estas etapas las cuales son:

- Interfaz de usuario.
- Control
- Actuadores

Dentro de estas etapas incluimos la parte mecánica y electrónica que será explicado a continuación.

FIGURA 26. Arquitectura del robot

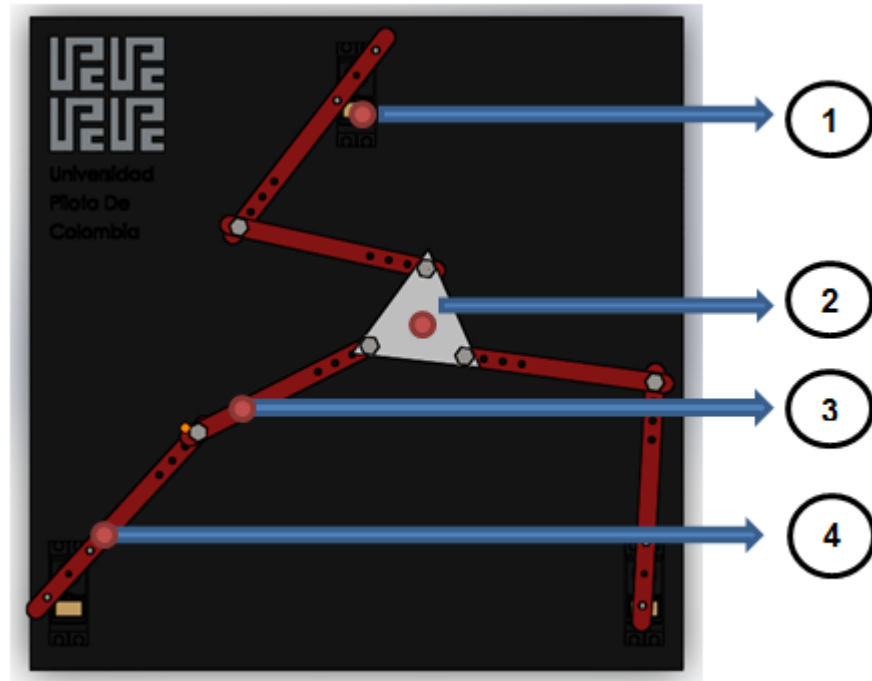


4.7 DISEÑO MECANICO

Usando una herramienta CAD teniendo en cuenta los costos de fabricación, eficiencia y reparación, se procedió a realizar el diseño pieza por pieza, idealmente se realizó el robot para fines didácticos y/o un pantógrafo digital, se aclaró lo anterior para considerar el tamaño necesario del robot y el material en el que se fabricó, en la figura 27, se da a conocer las partes que componen un robot Planar 3RRR.

- 1) Servo motor.
- 2) Efecto final, en este va ubicado un lápiz, o cualquier herramienta de dibujo similar.
- 3) Eslabones b_i , o junta pasiva.
- 4) Eslabones a_i , o junta activa.

Figura 27. Partes de un robot planar 3RRR



4.8 DIMENSIONES DEL PROTOTIPO

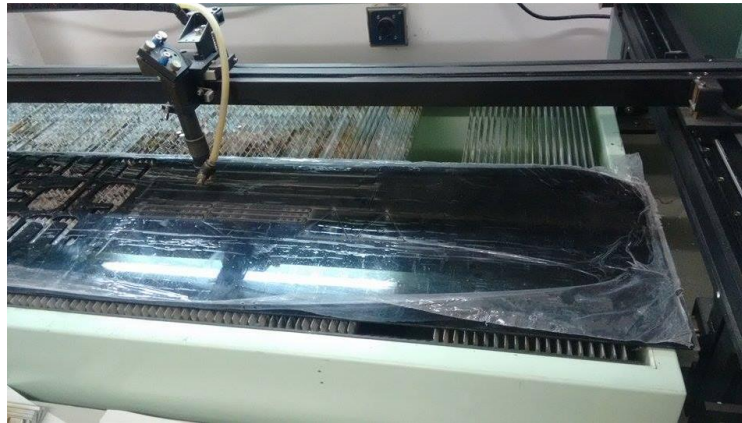
Ya que se trata de un prototipo didáctico y educativo no existe la necesidad que sea muy grande, como se había mencionado anteriormente los robots paralelos se caracterizan por no tener un reducido espacio de trabajo, y este caso no será la excepción, pero entre más grande será la distancia entre servos y la longitud de los eslabones sea la adecuada, el espacio de trabajo aumenta considerablemente.

El robot posee tres servomotores, estos están a una distancia de 30 cm cada uno formando un triángulo equilátero, cada uno de estos está conectado mediante pasadores a un primer eslabón, que tiene la posibilidad de cambiar de longitud, puede ser de 10 cm hasta 7 cm, el objetivo de esto es modificar y obtener mayor conocimiento acerca del espacio de trabajo, se conecta un segundo eslabón por medio de una junta rotacional, tiene las mismas características que el anterior eslabón, la diferencia es que el otro extremo está conectado a la plataforma móvil, que es un triángulo equilátero, cuyos lados poseen una longitud de 5 cm, las otras dos esquinas de la plataforma están acopladas a los dos eslabones de los servomotores faltantes, todo esto está apoyado en una base de 34 cm x 34 cm.

4.9 COMPONENTES MECÁNICOS DEL ROBOT

En primera instancia se realizó el diseño usando una herramienta CAD de cada una de las piezas que componen el prototipo, garantizando un diseño eficaz y versátil que sea capaz de suplir las necesidades que se han planteado, luego se enviaron las piezas a corte (Figura 28).

Figura 28. Corte de acrílico



4.9.1 ESLABONES (a_i) Estos son aquellos elementos que van acoplados al servomotor y a su vez al eslabón (b_i), tienen como principal característica que son de longitud variable dentro del conjunto del robot, desde 7cm hasta 10cm como se puede detallar en la (Figura 29).

Figura 29. Diseño eslabón a_i



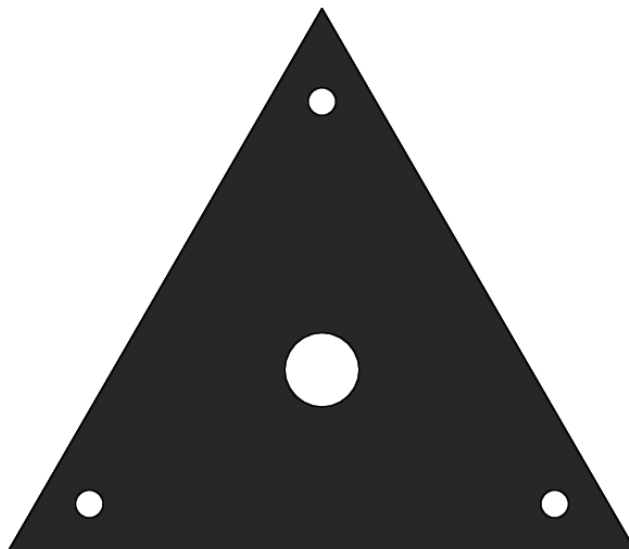
4.9.2 ESLABONES (b_i) Estos elementos (Figura 30) se acoplan a los eslabones (a_i) y a su vez al efector final, también tiene la posibilidad de cambiar su longitud de 7cm a 10cm

Figura 30. Diseño de eslabón b_i



4.9.3 PLATAFORMA MOVIL La plataforma móvil (Figura 31) es aquella que reúne todas las cadenas cinemáticas, su función para este caso es mostrar en que parte del plano está ubicado, tiene muchas funciones como alojar el efector final

Figura 31. Diseño de la plataforma móvil



4.9.4 BASE CON LOGO UNIVERSIDAD PILOTO DE COLOMBIA

La base (Figura 32) es la pieza donde van ubicados los servomotores y donde se encuentra el logo de la universidad

Figura 32. Diseño de la Base con logo de la universidad



4.9.5 BASE SOBRE BASE

Esta pieza (Figura 33) tal vez es la más sencilla pero una de las más importantes ya que soporta el peso de los servomotores, de los brazos y el efector final

Figura 33. Diseño de base sobre base



4.9.6 LATERALES DE LA BASE

Estos laterales (Figura 34) tienen unos pequeños agujeros rectangulares con el fin de que generen un espacio para cablear el sistema de robot

Figura 34. Diseño de laterales de la base



4.9.7 FRONTALES DE LA BASE

Su función (Figura 35) básicamente es separar las dos capas de las bases con el fin de no dejar ver los componentes como el arduino y cables, con el fin de optimizar presentación y el diseño del mecanismo

Figura 35. Diseño de frontales de la base



4.9.8 ENSAMBLAJE TOTAL DEL ROBOT

Finalmente en la (Figura 36) se puede detallar el prototipo completamente ensamblado usando las piezas anteriormente descritas y en la (Figura 37) se muestra la forma en la que se deben unir los componentes del robot planar RRR

Figura 36. Vista isométrica del robot



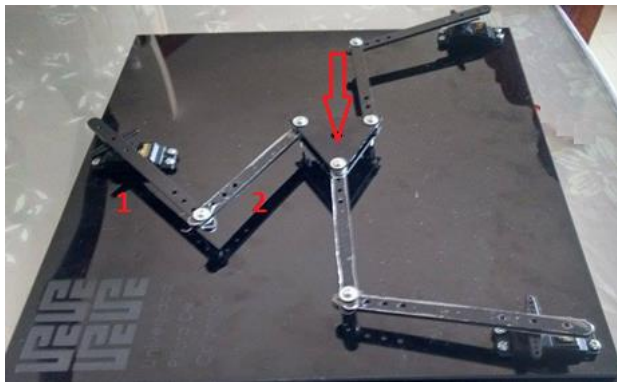
Figura 37. Vista explosionada



4.10 ANÁLISIS ESTÁTICO

En el diseño mecánico del robot se consideró de vital importancia un análisis a profundidad de las dimensiones de los eslabones y de la elección del material requeridos para obtener un excelente producto, para ello se consideró en primera instancia realizar un análisis estático del prototipo debido a que gracias al análisis jacobiano del sistema se puede complementar el análisis dinámico del mismo. De acuerdo a lo anterior, tomando como punto de referencia la figura 36 se plantea que se debe ejecutar un solo análisis de una de las cadenas cinemáticas porque las tres deben ser iguales, teniendo eso en cuenta se procede dimensionando las fuerzas que actúan sobre el sistema realizando el siguiente diagrama:

Figura 38. Tercera cadena cinemática del robot

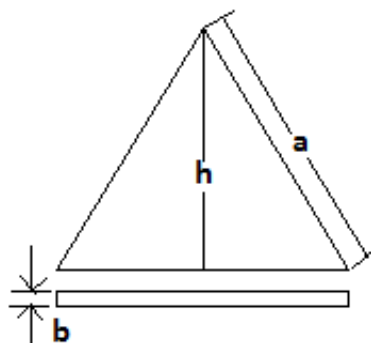


Observando la figura 38 se determina que la cadena cinemática seleccionada para efectuar el análisis del sistema, es la tercera. Para realizar los cálculos mecánicos requeridos en este proyecto, se consideró en primera instancia que

el robot estuviera estático, posteriormente se analizaron los eslabones del prototipo como si fuesen vigas teniendo en cuenta que en este caso su perfil es rectangular, también se consideró que por cuestión de costos era mejor fabricar todo el robot en el mismo material, debido a que elegir materiales diferentes podía requerir procesos de manufactura diferentes y adicionalmente el tiempo de producción también podía aumentar, se concluyó que la base móvil del robot se encuentra soportada en tres puntos, que son los extremos de cada cadena cinemática es decir los puntos A, B y C, razón por la cual para el análisis de esta cadena cinemática, se toma la fuerza ejercida por el peso de esta pieza y de la plataforma móvil como un tercio de su magnitud total.

Gracias a las consideraciones del problema enunciadas anteriormente, se puede proceder con el cálculo de las fuerzas que actúan sobre el robot, en primera instancia se calcularon los pesos de las piezas involucradas en el presente análisis, que son la base móvil y un eslabón debido a que el eslabón 1 y 2 según se puede observar en la (Figura 38) que son iguales, estos eslabones hacen referencia a los eslabones a y b de todo el sistema respectivamente. Suponiendo para los cálculos el mismo material de las piezas, en primera instancia para la plataforma móvil, tenemos que:

Figura 39. Esquemático superior base móvil



$$a = 50 \text{ mm} \rightarrow a = 5 \text{ cm}$$

$$b = 3 \text{ mm} \rightarrow b = 0.3 \text{ cm}$$

Se considera que la plataforma móvil es un triángulo equilátero (Figura 39), por lo que todos sus lados son de 5 cm. El área para la base móvil está definida por la fórmula (5.0):

$$A = \frac{\sqrt{3}a^2}{4} \rightarrow \tag{5.0}$$

$$A = \frac{\sqrt{3}(5 \text{ cm})^2}{4} \rightarrow A = 10.82 \text{ Cm}^2$$

Para la primera iteración del cálculo se asume como material el acrílico, debido a que es económico y el proceso de manufactura requerido también lo es, adicional a ello se necesita poco tiempo en su etapa de corte. Usando la

densidad (Ec.5.2) del material y el volumen (Ec.5.1) de la pieza se encuentra la fuerza que ejerce la pieza:

$$\rho = 1.18 \frac{g}{cm^3}$$

$$V = A \cdot b \tag{5.1}$$

$$V = 10.82 \text{ cm}^2 \cdot 0.3 \text{ cm} \rightarrow V = 3.247 \text{ cm}^3$$

$$\rho = \frac{m}{V} \rightarrow m = \rho \cdot V \tag{5.2}$$

$$m = 1.18 \frac{g}{cm^3} \cdot 3.247 \text{ cm}^3 \rightarrow m = 3.832 \text{ g} \rightarrow m = 0.003832 \text{ Kg}$$

Como la base móvil está compuesta por dos piezas con forma triangular la masa debe ser el doble y se debe considerar la masa de los elementos de sujeción y del efector final, que para este caso es un lápiz de 5.2 g:

$$m = (2 \cdot 3.832 \text{ g}) + 5.2 \text{ g} + 2.13 \text{ g} \rightarrow m = 14.994 \text{ g} \rightarrow m = 15 \text{ g} \rightarrow m = 0.015 \text{ Kg}$$

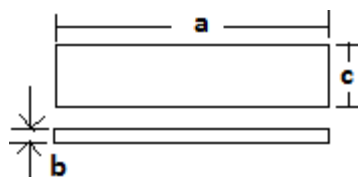
La fuerza (Ec.5.3) ejercida por la pieza y el efector final es:

$$F = mg \tag{5.3}$$

$$F = 0.015 \text{ Kg} \cdot 9.8 \frac{m}{s^2} \rightarrow F = 0.147 \text{ N}$$

Hallando el volumen y masa se pudo calcular la fuerza ejercida (Ec.5.3) por los eslabones, teniendo en cuenta las dimensiones (Figura 40):

Figura 40. Dimensiones de los eslabones



$$\begin{aligned}
 a &= 140 \text{ mm} \rightarrow a = 14 \text{ cm} \\
 b &= 3 \text{ mm} \rightarrow b = 0.3 \text{ cm} \\
 c &= 10 \text{ mm} \rightarrow c = 1 \text{ cm}
 \end{aligned}$$

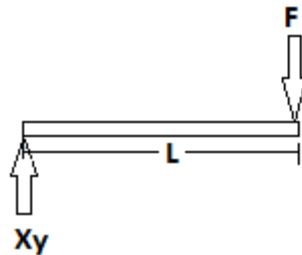
$$V = a \cdot b \cdot c \rightarrow V = 14 \text{ cm} \cdot 1 \text{ cm} \cdot 0.3 \text{ cm} \rightarrow V = 4.2 \text{ cm}^3$$

$$m = 1.18 \frac{\text{g}}{\text{cm}^3} \cdot 4.2 \text{ cm}^3 \rightarrow m = 4.956 \text{ g} \rightarrow m = 0.0049 \text{ Kg}$$

$$F_e = mg \rightarrow F_e = 0.0049 \text{ Kg} \cdot 9.8 \frac{\text{m}}{\text{s}^2} \rightarrow F_e = 0.048 \text{ N}$$

Realizando el diagrama de cuerpo libre (Figura 41) para el eslabón 2 de la cadena cinemática:

Figura 41. Diagrama de cuerpo libre para eslabones b



Según se había dicho anteriormente, la fuerza F en esta viga es un tercio de la fuerza ejercida por la base, es decir que la fuerza $F = 0.049 \text{ N}$. Hallando la reacción X_Y :

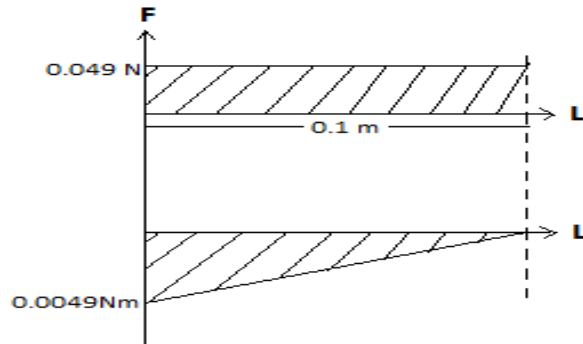
$$\begin{aligned}
 \sum F_X &= 0 \\
 \sum F_Y &= X_Y - F = 0 \rightarrow X_Y = F \rightarrow X_Y = 0.049 \text{ N}
 \end{aligned}$$

Hallando las ecuaciones de fuerza cortante y momento flector usando funciones de singularidad:

$$\begin{aligned}
 V &= X_Y \langle L - 0 \rangle^0 - F \langle L - 0.1 \rangle^0 \\
 M &= X_Y \langle L - 0 \rangle^1 - F \langle L - 0.1 \rangle^1
 \end{aligned}$$

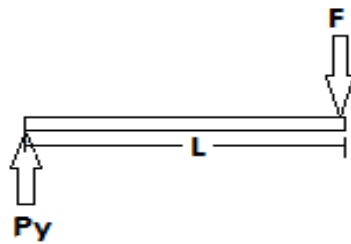
Graficando el diagrama de fuerza cortante y momento flector:

Figura 42. Diagrama de fuerza cortante y momento flector eslabón 2



Realizando el mismo análisis para el eslabón 1 (Figura 43):

Figura 43. Diagrama de cuerpo libre para eslabones a



En este caso la fuerza F es la suma entre la fuerza ejercida por el peso del eslabón 2 y la reacción del punto X hallada anteriormente:

$$F = F_e + X_Y \rightarrow F = 0.049 \text{ N} + 0.048 \text{ N} \rightarrow F = 0.097 \text{ N}$$

Aplicando el mismo análisis realizado al eslabón 2:

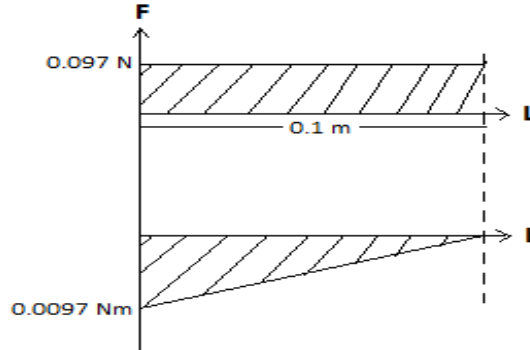
$$\sum F_X = 0$$

$$\sum F_Y = P_Y - F = 0 \rightarrow P_Y = F \rightarrow P_Y = 0.097 \text{ N}$$

$$V = P_Y \langle L - 0 \rangle^0 - F \langle L - 0.1 \rangle^0$$

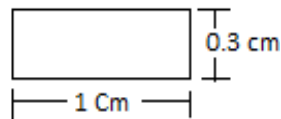
$$M = P_Y \langle L - 0 \rangle^1 - F \langle L - 0.1 \rangle^1$$

Figura 44. Diagrama de fuerza cortante y momento flector eslabón 1



De acuerdo a la (Figura 44) anterior, se concluye que el momento máximo aplicado sobre el eslabón 1 Es de 0.0097 Nm. Calculando el momento de inercia en X (Ec.5.4) del eslabón teniendo en cuenta el perfil (Figura 45):

Figura 45. Perfil del eslabón



$$I_x = \frac{b \cdot h^3}{12} \quad (5.4)$$

$$I_x = \frac{0.01 \text{ cm} \cdot (0.003 \text{ m})^3}{12} \rightarrow I_x = 22.5 \times 10^{-12} \text{ m}^4$$

$$C = 0.0015$$

Determinando el esfuerzo (Ec.5.5) al que se encuentra sometido:

$$\sigma = \frac{MC}{I} \quad (5.5)$$

$$\sigma = \frac{0.097 \text{ Nm} \cdot 0.0015 \text{ m}}{22.5 \times 10^{-12} \text{ m}^4} \rightarrow \sigma = 646666.66 \text{ Pa} \rightarrow \sigma = 646366 \text{ KPa}$$

Hallando el factor de seguridad (Ec.5.6):

$$FS = \frac{\sigma_Y}{\sigma} \quad (5.6)$$

$$FS = \frac{100 \text{ MPa}}{646366 \text{ KPa}} \rightarrow FS = 154.64$$

Basándose en los cálculos anteriores se puede afirmar, que para la carga a la que normalmente va estar sometido el robot que es la fuerza ejercida por el lápiz, el acrílico es un material excelente para esta aplicación bajo condiciones normales de trabajo, sin embargo teniendo en cuenta el factor de seguridad, se determina que para la carga anteriormente descrita el material está sobredimensionado. Es prudente determinar la carga máxima que el robot es capaz de soportar usando como material el acrílico con el fin de conocer una de las principales limitaciones del prototipo, proporcionándole al operador información veraz sobre el funcionamiento del equipo protegiendo la integridad del mismo.

Para calcular el límite de resistencia por flexión de los eslabones, es necesario calcular el esfuerzo máximo (Ec.5.5) que son capaces de soportar, para esto se realizaron los cálculos con un factor de seguridad (Ec.5.6) de 1:

$$FS = \frac{\sigma_Y}{\sigma} \rightarrow \sigma = \frac{\sigma_Y}{FS} \rightarrow \sigma = \frac{100 \text{ MPa}}{1} \rightarrow \sigma = 100 \text{ MPa}$$

$$\sigma = \frac{MC}{I} \rightarrow M = \frac{\sigma I}{C} \rightarrow M = \frac{(100 \text{ MPa})(22.5 \times 10^{-12} \text{ m}^4)}{0.0015 \text{ m}} \rightarrow M = 1.5 \text{ Nm}$$

El momento máximo que soporta el eslabón antes de fallar es de 1.5 Nm, reemplazando este valor de la sumatoria de momentos en el punto P se calcula la fuerza que soporta el material:

$$\sum M_p = M - F(0.1 \text{ m}) = 0 \rightarrow F = \frac{M}{0.1 \text{ m}} \rightarrow F = \frac{1.5 \text{ Nm}}{0.1 \text{ m}} \rightarrow F = 15 \text{ N}$$

$$F = mg \rightarrow m = \frac{F}{g} \rightarrow m = \frac{15 \text{ N}}{9.8 \frac{\text{m}}{\text{s}^2}} \rightarrow m = 1.53 \text{ Kg}$$

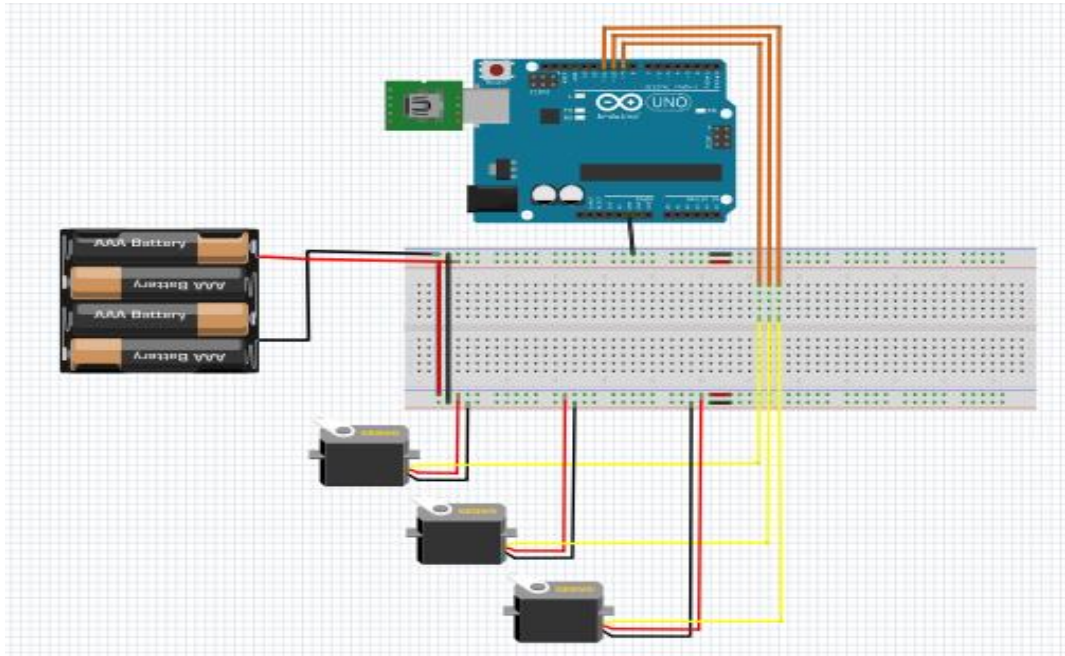
De acuerdo a lo anterior la fuerza máxima que soporta cada uno de los eslabones es de 15 N, es decir que si se le coloca un elemento de más de 1.53 Kg sobre la pieza, esta fallará. Finalmente se comprueba que el acrílico es útil para esta aplicación y las dimensiones de los eslabones y de la plataforma móvil se consideran acertadas.

4.11 DISEÑO ELECTRONICO

Se procede a realizar se diseñó electrónico, lo primero que se considera es con que sistema mover los eslabones, también con que se enlazará el software del

computador con los actuadores en la (Figura 46), se mostrara la conexión y la arquitectura del sistema electrónico.

Figura 46. Conexión electrónica



4.11.1 Arduino Existen muchos métodos de control de actuadores, para ello se debe considerar muchos aspectos, los cuales son indispensables a la hora de hacer el diseño, inicialmente se había escogido realizarlo con un PIC, ya que tiene muchas más aplicaciones que arduino, era uno de los más vendidos ya que su fabricante es MICROCHIP y es una marca posicionada en el mercado, su costo es accesible, está entre los 15000 y 20000 pesos colombianos, se puede descargar su software y programar en MPLAB, su instalación es un poco tediosa ya que toca hacer un circuito, alimentarlo correctamente y a su vez hacer otro circuito para la conexión con el computador, he aquí donde surgió la necesidad de buscar otra opción para controlar los actuadores que facilitara el montaje y generara una reducción de costos.

Para realizar el control del sistema se escogió el sistema Arduino UNO debido a que las características que posee este elemento eran las más apropiadas en cuanto a funcionalidad y costo, esta tarjeta es la encargada de interactuar en tiempo real con la interfaz gráfica desarrollada en MATLAB, sirviendo de puente de comunicación entre el software programado y los actuadores, que para este caso son servomotores. Arduino UNO establece la comunicación por puerto serial entre el computador que posee el software y los actuadores del sistema, gracias a este elemento se pueden controlar los tres servos de forma simultanea usando tres de sus salidas del tipo PWM. Con cada modo de uso empleado por el usuario en la interfaz gráfica, se obtienen los ángulos requeridos para situar la plataforma móvil dentro del plano correspondiente a

su área de trabajo, Dentro del código se efectúan las operaciones de cinemática inversa obtenidas gracias a la geometría del mecanismo, se visualizan los resultados dentro de la interfaz, se simula el movimiento del prototipo y se envía por medio de la comunicación cableada los ángulos obtenidos al ARDUINO UNO, que es el encargado de decodificar el valor requerido del ángulo, en un ancho de pulso determinado, que es transmitido a cada uno de los servomotores, logrando así que cada uno rote y se ubique en el ángulo establecido por el programa.

Arduino es una plataforma de adquisición de datos, el cual tiene código abierto ⁴⁴ y está basada en una placa cuyo microcontrolador es un atmel avr, tiene múltiples entradas y salidas las cuales pueden ser analógicas y digitales, poseen un software propio y su lenguaje de programación es C/C++, Existe diferentes tipos de arduino, para este proyecto se usó el Arduino uno (Figura 47), debido a su reducido costo y accesequibilidad, aparte de su sencillo manejo.

Figura 47. Arduino uno



Para poder trabajar con arduino tenemos que conocer la placa como tal (tabla 2), ya que cada placa tiene diferentes especificaciones ⁴⁵ y diferentes propósitos.

-
- ⁴⁴ Proyecto Arduino ¿Qué es arduino?, [en línea] <<https://proyectoarduino.wordpress.com/%C2%BFque-es-arduino/>> [citado en 24 de agosto de 2015]
 - ⁴⁵ ARDUINO. ARDUINO /GENUINO UNO <<https://www.arduino.cc/en/Main/ArduinoBoardUno>>[citado en 26 de agosto de 2015]

Tabla 2. Especificaciones técnicas arduino uno

Microcontrolador	Atmega328P
Tensión de funcionamiento	5V
Voltaje de entrada(Recomendado)	7-12V
Voltaje de entrada(Limite)	6-20V
Pines digitales I/O	14(de los cuales 6 proporcionan salida de PWM)
Pines digitales PWM I/O	6
Pines de entradas análogas	6
Corriente continua por pines I/O	20mA
Corriente continua para pines de 3.3V	50mA
Memoria flash	32KB(ATmega328P) de los cuales 0.5KB son usados por el gestor de arranque
SRAM	2KB(ATmega328P)
EEPROM	1KB(ATmega328P)
Velocidad del reloj	16 MHz
Largo	68.6 mm
Ancho	53.4 mm
Peso	25g

4.11.2 Servomotor Un servomotor ⁴⁶es un motor de corriente continua acoplado mediante engranajes a un potenciómetro el cual permite conocer la posición y así poder obtener el ángulo deseado .Existen diferentes tipos de servos, algunos alcanzan 90, 180, y 360 grados ,a mayor grado mayor costo.

Estos motores son ideales para aplicaciones que requieran cambios de posición, estos elementos son útiles en el campo de la robótica debido a su precisión, funcionalidad y costo. Para este prototipo se emplearon 3 servomotores de características iguales que se describirán a continuación. La función de estos motores es proporcionar control al usuario sobre tres articulaciones, las cuales se han llamado articulaciones activas, para

⁴⁶ Sastre,Toni Ruiz ELECTROINSAIMADA TUTORIALES RASPBERRY PI , ARDUINO ,ELECTRONICA Y MUCHO MAS. Tutorial Arduino : Servomotor, <<http://www.electroensaimada.com/servomotor.html>> [citado en 24 agosto 2015]

controlarlos es necesario usar salidas de tipo PWM. Para esta aplicación se pueden usar servomotores de giro de 0-180°, gracias al código programado que revierte el ángulo calculado si llega a ser mayor a 180°, es decir que usará la ecuación alterna de obtención del ángulo, recordando que cada ángulo tiene dos posibles soluciones. El controlador empleado para estos motores es el ARDUINO UNO descrito anteriormente, gracias a esta placa se puede traducir el ángulo calculado y enviado por puerto serial desde MATLAB, hacia el servomotor en un pulso de longitud de onda variable en función del ángulo requerido, permitiendo al operario tener total control de los servomotores garantizando que el ángulo requerido y calculado por el software es el que ejecutarán los servomotores.

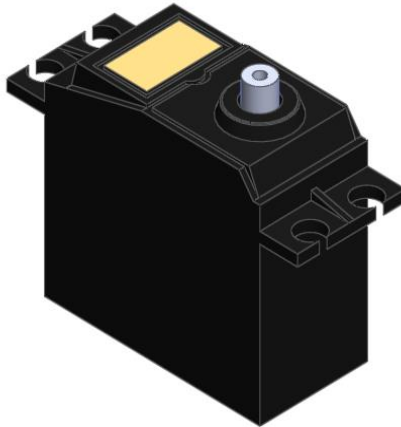
4.11.2.1 Servomotor sg-5010 Este como cualquier componente mecánico y electrónico tiene unas características (Tabla 3) que es necesario conocerlas antes de comenzar cualquier proyecto.

Tabla 3. Características servo SG-5010

Modulación	Digital
Torque	4.8V(5.5 Kg-cm) 6.0V(6.50 Kg-cm)
Velocidad	4.8V (0.19 sec/60º) 6.0V (0.15 sec/60º)
Peso	47.0 g
Dimensiones	Largo (46.0 mm) Ancho (20.5 mm) Altura (38.0 mm)
Tipo de motor	3-pole
Tipo de engranaje	plástico
Rotación /apoyos	Cojinetes dobles

Cuando obtiene su mayor voltaje, el servomotor opera a su máxima capacidad de torque y de velocidad, el servo fue modelado (Figura 48) en una herramienta CAD al igual que todas las piezas.

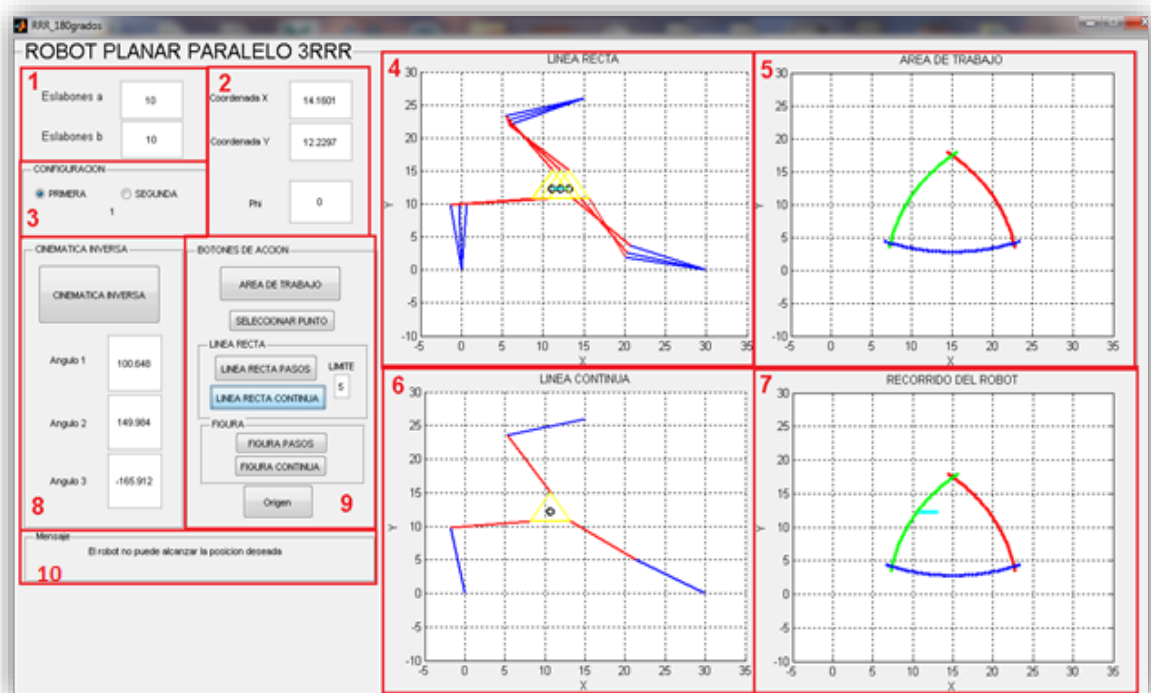
Figura 48. Modelado del servo SG-5010



4.12 INTERFAZ DE USUARIO

En la figura 49 se puede observar la interfaz gráfica, que se realizó, por medio de la cual se puede interactuar en tiempo real con el prototipo, y es el medio que nos permite crear un puente entre el hombre y la máquina para poder generar el movimiento que deseemos que el prototipo efectúe.

Figura 49. Interfaz grafica



A continuación se listan y describen cada uno de los componentes que conforman la interfaz:

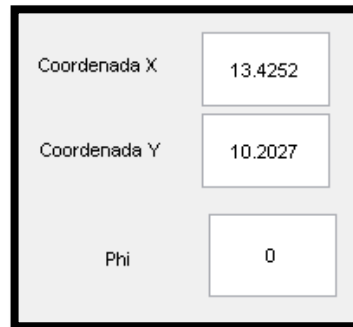
1) Longitudes: En esta sección se puede modificar las longitudes de los eslabones (en centímetros) según se desee, en este caso en el ejemplo se introdujeron 10 cm (Figura 50) debido a que físicamente así se diseñaron cada uno de los eslabones, hay que tener en cuenta que el recuadro *eslabones a*, establece la misma longitud para los eslabones a_1 , a_2 y a_3 , y lo mismo sucede con el recuadro *eslabones b*.

Figura 50. Ítem 1, Longitudes

Eslabones a	10
Eslabones b	10

2) Coordenadas: En esta sección (Figura 51) se introduce el ángulo de orientación de la plataforma móvil y la coordenada en la que se desea que el efector final (Centro del triángulo equilátero, punto negro según sección 4) se ubique, se puede introducir de forma manual o automática.

Figura 51. Ítem 2, Coordenadas

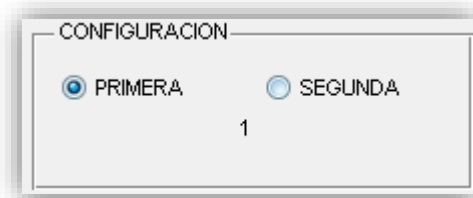


A screenshot of a software interface for entering coordinates. It features three input fields arranged vertically. The first field is labeled 'Coordenada X' and contains the value '13.4252'. The second field is labeled 'Coordenada Y' and contains the value '10.2027'. The third field is labeled 'Phi' and contains the value '0'.

Coordenada X	13.4252
Coordenada Y	10.2027
Phi	0

3) Configuración: Como se enunciaba anteriormente el robot puede tener 8 soluciones para llegar al punto que se introduzca (2 por cada ángulo), si se selecciona la *configuración primera* el robot se ubicara de una forma, es decir cada ángulo θ_1, θ_2 y θ_3 , se obtendrá la primera solución, y si se selecciona la *configuración segunda*, los ángulos darán como resultado su segunda solución (Cabe recordar que cada ecuación de cada ángulo es de segundo grado, por lo que posee dos soluciones).

Figura 52. Ítem 3, Configuración



A screenshot of a configuration dialog box titled 'CONFIGURACION'. It contains two radio button options: 'PRIMERA' (which is selected) and 'SEGUNDA'. Below these options, the number '1' is displayed.

CONFIGURACION	
<input checked="" type="radio"/> PRIMERA	<input type="radio"/> SEGUNDA
1	

4) Grafica robot por pasos: En esta parte se puede visualizar (Figura 53) el comportamiento del robot en tiempo real, ya sea como una imagen o como una simulación según la opción que el usuario desee, en esta grafica el robot ejecutará movimientos con pocas muestras por lo que las simulaciones se observaran en forma de pasos.

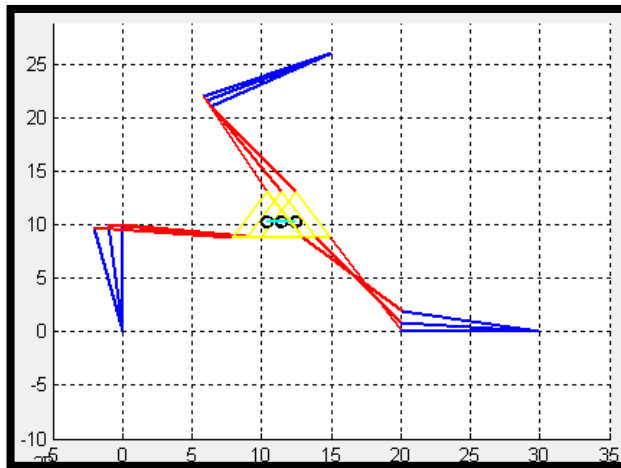
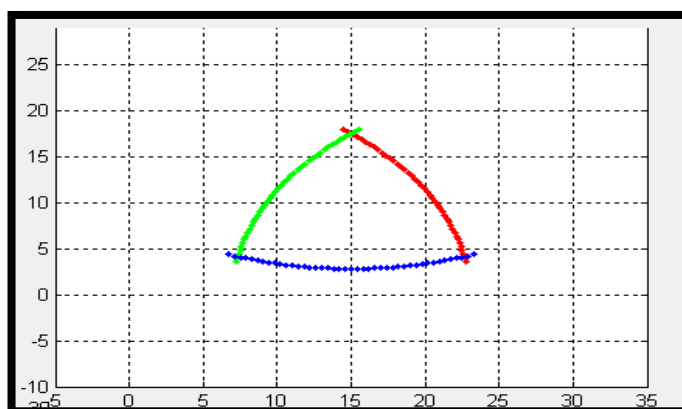


Figura 53. Ítem 4, Gráfica del robot por pasos

5) Grafica área: En esta sección se puede ver el área de trabajo (Figura 54) que el robot es capaz de alcanzar, es decir que cualquier punto que este fuera del área de ejemplo no podrá ser alcanzado por las limitaciones físicas del prototipo

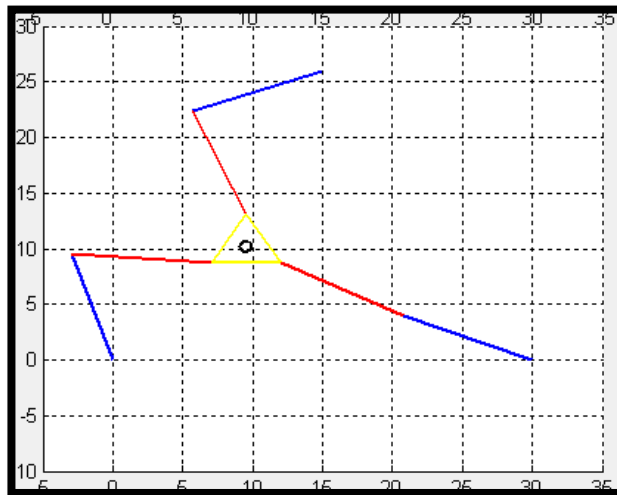
Figura 54. Ítem 5, Área de trabajo



6) Grafica robot continua: A diferencia de la sección 4, en esta sección se grafica solo las simulaciones de movimiento en tiempo real del prototipo que el usuario desee en la sección 8 botones de acción. En esta área (Figura 55) se

simula el sistema con una gran cantidad de muestras por lo que el movimiento del robot se suaviza y se vuelve continuo.

Figura 55. Ítem 6, Grafica robot Continua



7) Recorrido: En la sección de recorrido en la cual se muestra el trayecto recorrido por el prototipo en cualquiera de sus modos de simulación, por ejemplo: la (Figura 56) denota la trayectoria recorrida por el robot.

Figura 56. Ítem 7, Recorrido del robot



8) **Botón Cinemática Inversa:** En esta sección se encuentra el botón *CINEMÁTICA INVERSA* (Figura 57) el cual permite al programa ejecutar las ecuaciones programadas tomando como datos de entrada los datos de las secciones *longitudes y coordenadas*, adicionalmente en esta sección se muestran los resultados de los ángulos θ_1, θ_2 y θ_3 , de arriba hacia abajo.

Figura 57. Ítem 8, Cinemática Inversa

CINEMATICA INVERSA

CINEMATICA INVERSA

Angulo 1	107.45
Angulo 2	156.915
Angulo 3	-158.526

9) **Botones de acción:** Por medio de estos botones (Figura 58) se puede seleccionar la opción de movimiento o función que se desee. El botón *ÁREA DE TRABAJO* permite hallar el área de trabajo del prototipo que se visualizara

en la sección 5, anteriormente enunciada. Posteriormente nos permitirá seleccionar cualquier punto dentro de esta sección para que se pueda guardar en la sección 2 *Coordenadas*.

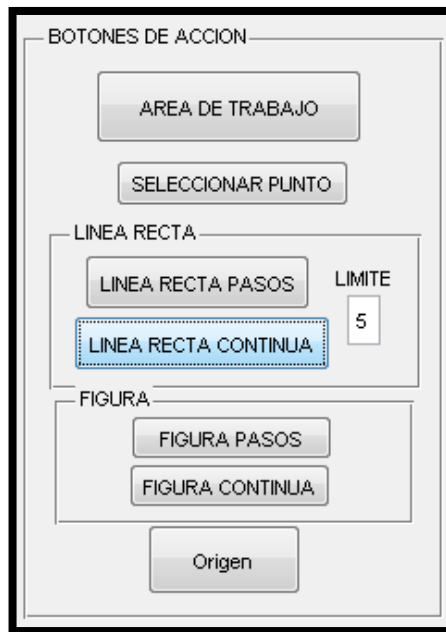
El botón *SELECCIONAR PUNTO*, permite seleccionar una coordenada de la sección 5 sin calcular el área de trabajo, que es la principal diferencia con el botón *Área de trabajo*.

En la subdivisión *LÍNEA RECTA*, aparece el botón *LÍNEA RECTA PASOS* el cual genera un movimiento solo en el eje X es decir que mantiene la coordenada en Y constante con un intervalo de muestreo bajo, por lo que la simulación se efectúa en pasos grandes, el movimiento se realiza de derecha a izquierda, las coordenadas previamente introducidas son el punto de inicio para dicho movimiento lineal del efector final y el punto máximo en X se introduce en la casilla *LIMITE*, también se encuentra en esta subdivisión el botón *LÍNEA RECTA CONTINUA* que básicamente efectúa el mismo proceso que el botón anterior, pero tomando una mayor cantidad de muestras, que tienen como consecuencia un movimiento más armónico y la cantidad de pasos es muy pequeña, al finalizar cualquiera de los dos botones en la sección 7 *Recorrido* aparecerá la trayectoria realizada por el efector final .

En la subdivisión *FIGURA*, aparece el botón *FIGURA PASOS* que permite seleccionar varios puntos que se deseen del área de trabajo (Sección 5) y al oprimir el botón *ENTER* el robot ejecuta los puntos que el usuario definió, de tal forma que el robot dibuja cualquier figura que el operario desee con un intervalo de muestreo grande por lo que la simulación del movimiento se visualiza en forma de pasos. En la misma subdivisión está el botón *FIGURA CONTINUA* que efectúa el mismo proceso que el botón anterior pero con un número de muestras mayor, al igual que el botón *LÍNEA RECTA CONTINUA* la simulación se suaviza y se puede observar de forma más eficiente el comportamiento real del prototipo, al finalizar cualquiera de los dos botones en la sección 7 *Recorrido* aparecerá la trayectoria realizada por el efector final

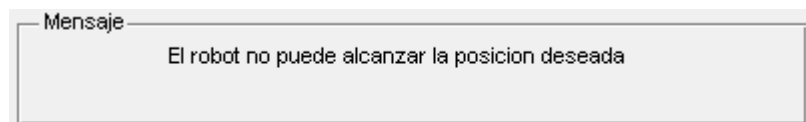
Finalmente el botón *ORIGEN* configura los servos en su posición inicial es decir 0° y configura los parámetros de comunicación entre Arduino y Matlab.

Figura 58. Ítem 9, Botones de acción.



10) Mensaje: En esta sección se muestra un mensaje de error (Figura 59) en dado caso que el prototipo físicamente no pueda alcanzar el punto que el usuario introdujo en la interfaz

Figura 59. Ítem 10, Mensaje de error



4.12.1 Comunicación matlab, arduino y servomotor A partir de la innovación de los usuarios de Matlab y Arduino, se creó una herramienta llamada Arduino IO, Este es un paquete que permite manejar la placa de arduino desde Matlab, sin importar el tipo de datos, analógico y digital, tampoco si recibe o envía los mismos. A continuación se darán a conocer los pasos para poder utilizar esta herramienta tan útil, hay que aclarar que este paquete esta patentado por el mismo MathWorks y al descargar el .zip están las instrucciones, pero para mayor facilidad se hará una lista de pasos y se darán a conocer en este documento⁴⁷, también se realizó un pequeño ejemplo del manejo de un servomotor.

• ⁴⁷ Arduino y Matlab Tutorial 01 Como Configurar arduino y matlab para realizar... IngJulianVelasco. <<http://www.youtube.com/watch?v=TitVKsC9tX0>> [citado en 18 de julio de 2015]

4.12.2 Instrucciones para instalar arduino io Los pasos necesarios para instalar arduino IO son

- Descargar Arduino IO en <http://www.mathworks.com/matlabcentral/fileexchange/32374-matlab-support-package-for-arduino-aka-arduinoio-package?download=true>
- Tener instalado Arduino y una versión superior a Matlab 2010 a.
- Abrir el software de Arduino y conectar la placa, que en este caso será Arduino uno.
- Establecer en que puerto (COM X) quedo la placa.
- Ir a Arduino, herramientas, tarjeta y escoger el puerto del punto anterior.
- Dentro de Arduino, tenemos que cargar a nuestra placa un sketch que se encuentra en la carpeta descargada Arduino IO, pde, adiosrv, adiosrv.pde.
Este sketch nos permite que la placa comprenda los datos enviados por Matlab.
- Se cierra el entorno de Arduino, ya que si se deja abierto y se inicia Matlab, Emite un mensaje de error debido a que dos objetos utilizan el mismo puerto.
- Ejecutar Matlab como administrador y en la carpeta actual se selecciona la carpeta descargada Arduino IO.
- En la ventana de comandos de Matlab , escribir “install_arduino” , para instalar
- Finalmente se escribe también en la ventana de comandos “a=arduino('COMX)”. La (X) se reemplaza por el número del COM.

4.12.3. Manejo de servomotores a traves de arduino io En este punto se explica de la configuración de un servomotor SG-5010 a través de Matlab y arduino, por medio de la herramienta Arduino IO. Inicialmente se requiere realizar los tres últimos pasos del punto anterior, tal como se explicó se debe ubicar la carpeta de Arduino IO (figura 60), luego se procede con la instalación con “install_arduino” (figura 61) y por último se configura el puerto escribiendo “a=arduino('COMX)”, COMX es el mismo puerto que se utilizó en arduino.

Figura 60. Ubicando carpeta Arduino IO

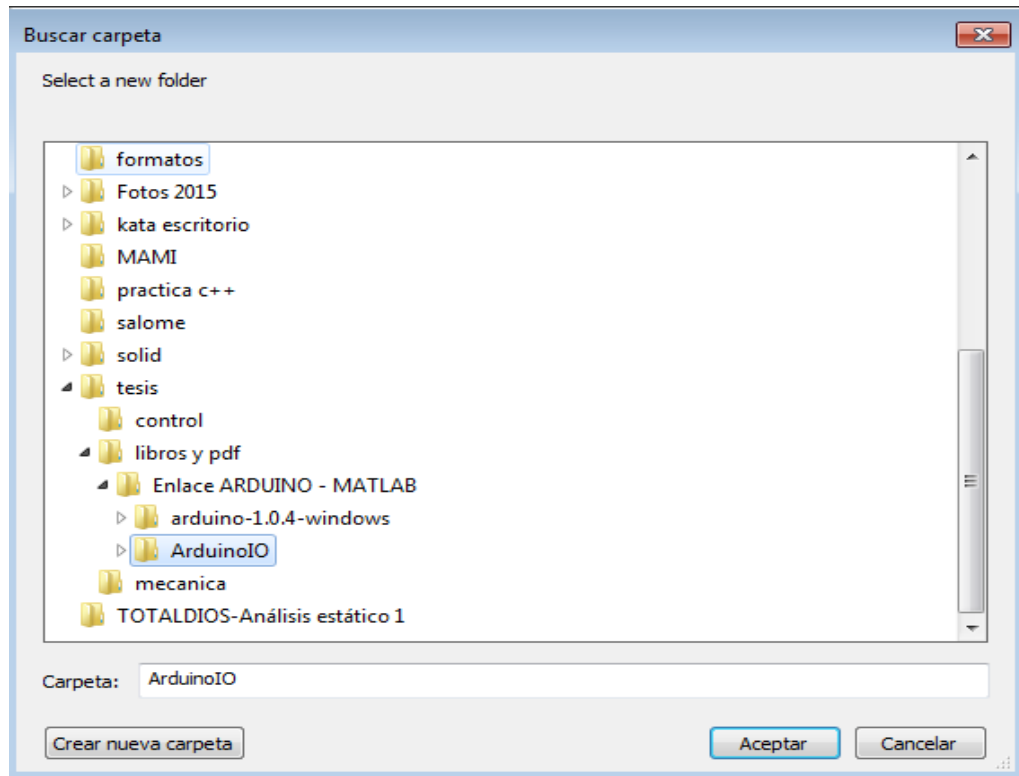


Figura 61. Instalación de Arduino IO

```
Command Window
>> install_arduino
Arduino folders added to the path
Saved updated MATLAB path

fx >> a=arduino('COMX')
```

Se probó el programa digitando el siguiente código en un editor de texto o en un archivo .m, este es un programa para mover un servomotor de (0 a 180) grados, en un intervalo de dos en dos y que se devuelva de la misma manera.

```
a.pinMode(13, 'output')
x = 0; %almacenamos la posicion en esta variable
max = false; %en este decimos si hay que sumar o restar
while true %mientras sea verdadero realice estas tareas
a.servoWrite(3,x) %enviamos el primer angulo al servo
if max %si el angulo ya es 150 es maximo
```

```

x = x-2           %restamos de dos en dos el angulo
a.digitalWrite(13,1); % encendemos un led
pause(0.5)       %hacemos una pausa de medio segundo
a.digitalWrite(13,0); % apagamos el led
pause(0.5)       %hacemos una pausa de medio segundo
else             %si no es maximo
x = x+2           %sumamos de dos en dos
a.digitalWrite(13,1); %encendemos un led
pause(0.5)       %pausa de medio segundo
a.digitalWrite(13,0); % apagamos el led
pause(0.5)       %pausa de medio segundo
end

if x == 150      %pregunta si es 150
max = true      % si el maximo es 150 reste de dos en dos hasta llegar
a 10
elseif x == 10 %pregunta si es 10 lo minimo
max = false     %si el maximo es 10 sume de dos en dos hasta llegar a
150
end

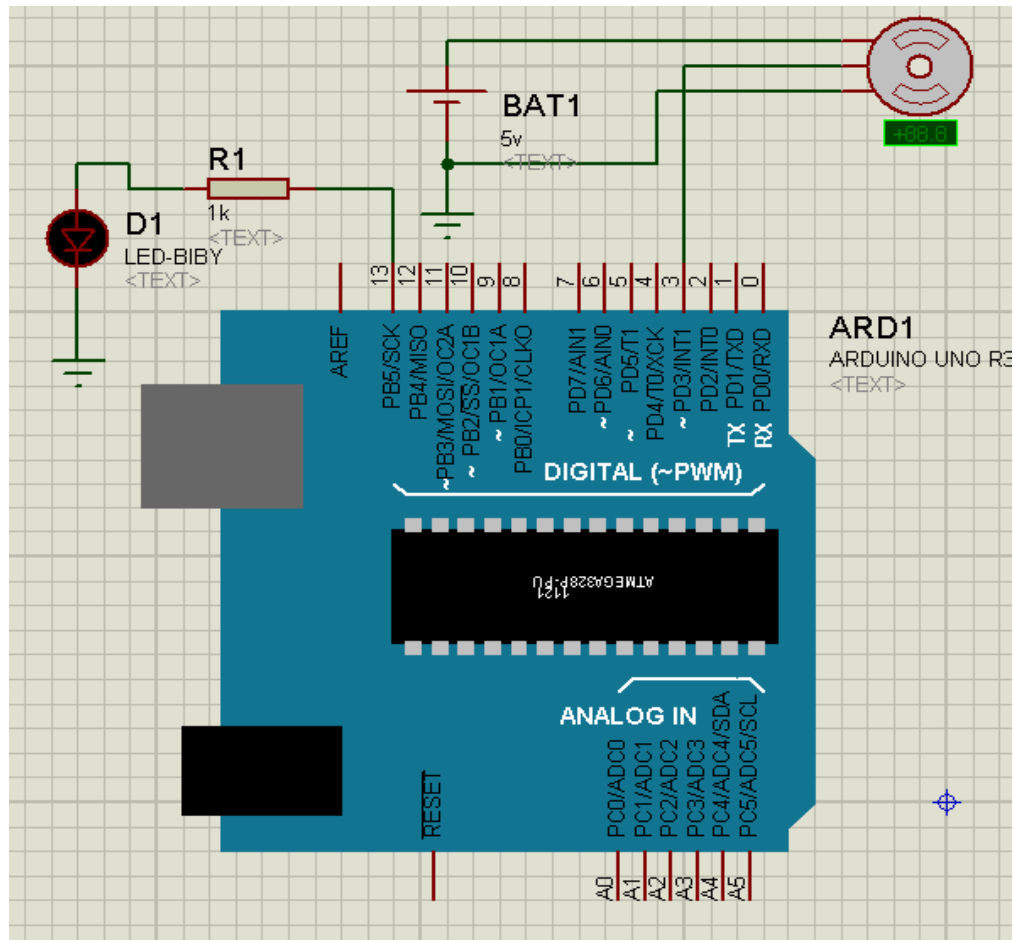
pause(0.5)

end

```

Como es una prueba no es necesario hacerlo en físico, pero se puede simular (figura 62) usando un software de electrónica como herramienta.

Figura 62. Conexión servomotor-arduino



Para que funcione el servomotor, este tiene que tener una alimentación externa dependiendo de las características del actuador, también cabe resaltar que obligatoriamente hay que acoplar la tierra de la fuente y del arduino.

5 ANALISIS DE RESULTADOS

5.7 PROTOTIPO REAL

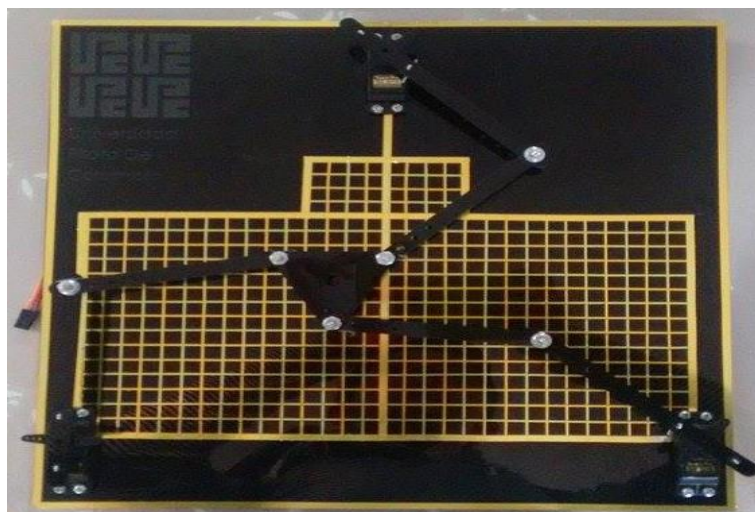
Luego de realizar el análisis y el diseño del mecanismo en la herramienta CAD, y posterior a esto, al corte del acrílico, se procedió a ensamblar las piezas, luego a colocar cada servo en su lugar y acoplarlos con los eslabones, en la figura 63 se puede detallar el robot completamente ensamblado.

Figura 63. Prototipo terminado



Completando el ensamblaje del robot, se introdujo una cuadrícula en cinta la cual tiene dos propósitos, conocer en qué posición se encuentra la plataforma móvil y la segunda mejorar la estética del prototipo, en la figura 64 se muestra el proyecto completamente listo para ser usado.

Figura 64. Prototipo con decorado



5.8 DISTINTOS ESPACIOS DE TRABAJO

Dadas las características del prototipo y las ecuaciones que definen su movimiento, es importante realizar un análisis a profundidad de sus características mecánicas y como estas afectan el desempeño del prototipo. Como se había mencionado con anterioridad las ecuaciones que describen el comportamiento y modelamiento del sistema se encuentran en función de variables independientes y dependientes, para el caso de la cinemática inversa encontramos que las variables independientes son:

$$a_i, b_i, X_Q, Y_Q, X_R, h, Y_R, X_P, Y_P, X, Y$$

Donde h es la longitud de cada uno de los lados del triángulo equilátero que es la base móvil y que por diseño es de 5 cm. a_i, b_i Son las longitudes de los eslabones para cada una de las cadenas cinemáticas. X, Y Son las coordenadas del efector móvil ubicado en el centro de la plataforma móvil. Las variables $X_Q, Y_Q, X_R, Y_R, X_P, Y_P$ son las coordenadas de cada una de las tres articulaciones sobre las que se ha realizado control (articulaciones activas), teniendo en cuenta que la coordenada X_P, Y_P es (0,0) porque la primera articulación rotacional se ubicó estratégicamente sobre el origen del plano que se usó para determinar la geometría del mecanismo. En el caso de las variables dependientes, se puede evidenciar en las ecuaciones que son

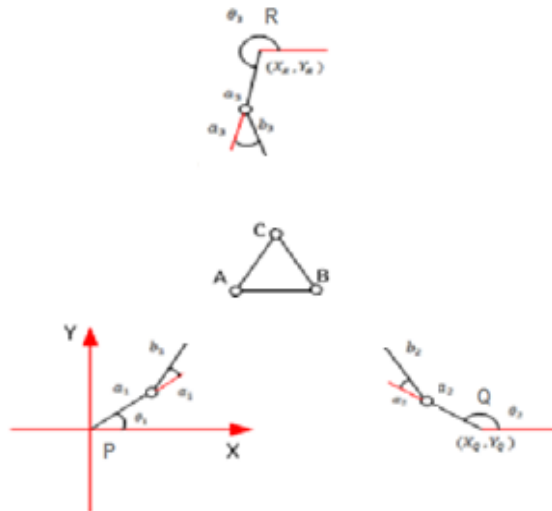
$$X_A, Y_A, X_B, Y_B, X_C, Y_C$$

Las Variables anteriores son las coordenadas del punto final de cada una de las cadenas cinemáticas es decir las coordenadas de los puntos A, B y C como se puede evidenciar en la (Figura 18).

De lo anterior se deduce que las variables que hacen referencia a características netamente mecánicas del prototipo son variables independientes, lo que permite que parametrizando las ecuaciones de movimiento se puedan variar con facilidad las diferentes respuestas del sistema. Con el fin de comprender de forma efectiva la relación que existe entre las longitudes de los eslabones del robot y su área de trabajo respectiva, se ha dedicado esta sección a realizar una comparación entre las áreas de trabajo que se podrían alcanzar con diferentes longitudes de eslabones, para este fin se usaron longitudes de 5 cm, 7.5 cm y 10 cm. Es importante resaltar que en cada caso, el valor de longitud se aplicó a todos los eslabones del prototipo y no a algunos de ellos, esto con el fin de mantener la misma forma de área de trabajo que posee el prototipo diseñado y que físicamente está construido con eslabones de 10 cm de longitud.

Debido a que las longitudes de los eslabones a_i, b_i se cambiaron a 5 cm (Figura 65), las coordenadas de cada una de las articulaciones Q, R debe cambiar debido a que físicamente los eslabones no se pueden unir ya que son de una longitud menor a las diseñadas que son de 10 cm, a excepción del punto P ya que siempre va estar ubicado en el origen como se puede ver en la siguiente imagen:

Figura 65. Cadenas cinemáticas con longitud de eslabones de 5 cm



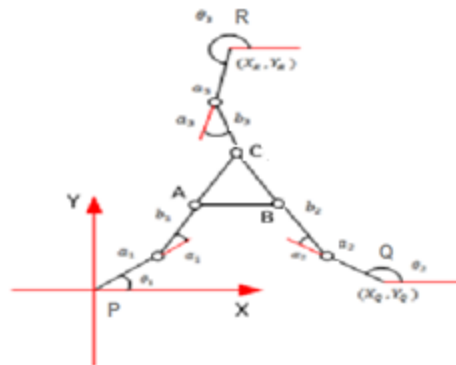
Para solucionar el problema anterior, las coordenadas en Q y R de las articulaciones rotacionales (Servomotores 2 y 3) se modificaron a:

$$(X_Q, Y_Q) = (15,0)$$

$$(X_R, Y_R) = (7.5, 12.97)$$

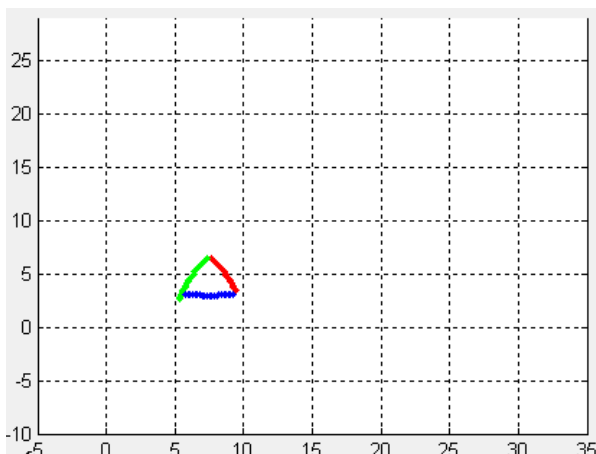
El resultado obtenido al aplicar el anterior cambio es (Figura 66):

Figura 66. Cadenas acopladas con eslabones de 5cm



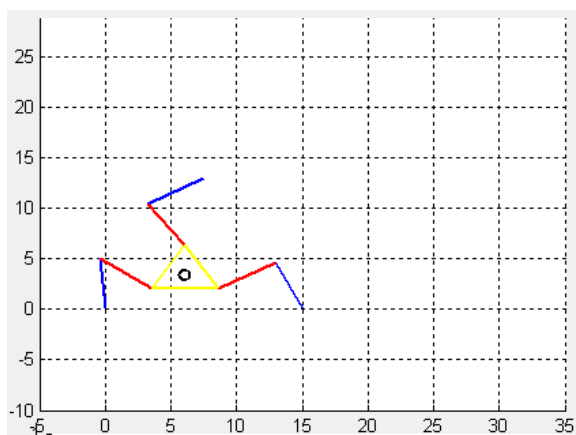
En el primer caso se realizaron las simulaciones de movimiento con eslabones de 5 cm de longitud, las coordenadas introducidas para el efector final fueron (6.07612, 3.44595) y el Angulo θ que corresponde a la orientación de la base móvil fue de 0° . El área de trabajo (Figura 67) obtenida fue:

Figura 67. Área de trabajo con eslabones de 5 cm



En la figura anterior se decidió mantener las mismas dimensiones del eje de graficación, con el fin de visualizar las diferencias entre las áreas de trabajo de las longitudes a comparar, estos límites de graficación se programaron para coordenadas mínimas de -10 cm en Y, -5 cm en X y las coordenadas máximas fueron 35 cm en X y 28 cm en Y. La posición (Figura 68) del robot para alcanzar las coordenadas introducidas previamente fue:

Figura 68. Prototipo con eslabones de 5 cm en configuración 1



Para este análisis se realizaron las simulaciones respectivas para configuración 1, es decir cuando las ecuaciones de cinemática inversa son positivas, recordemos que al ser una formula cuadrática, darán dos posibles soluciones por cada ángulo

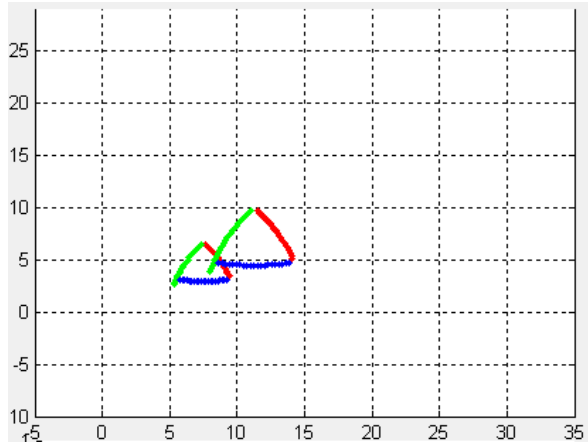
En el segundo caso se realizaron las simulaciones de movimiento con eslabones de 7.5 cm de longitud, para este caso las coordenadas articulares en Q y R fueron:

$$(X_Q, Y_Q) = (22.5, 0)$$

$$(X_R, Y_R) = (11.25, 19.46)$$

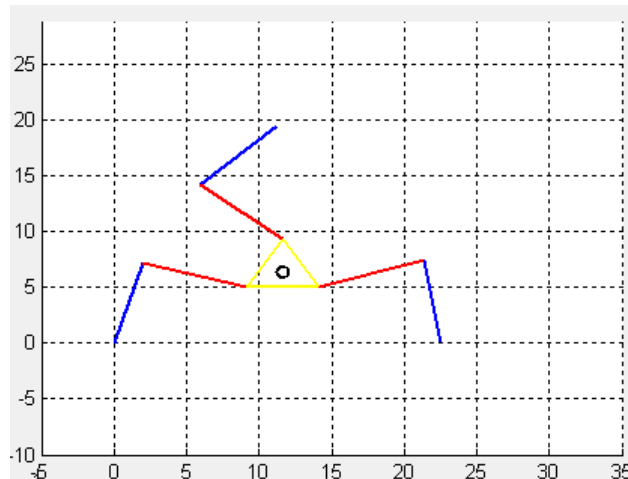
Las coordenadas introducidas para el efector final fueron (11.6404,6.41892) y el Angulo \emptyset que corresponde fue de 0° . El área de trabajo (Figura 69) obtenida fue:

Figura 69. Área de trabajo con eslabones de 5cm y 7.5 cm



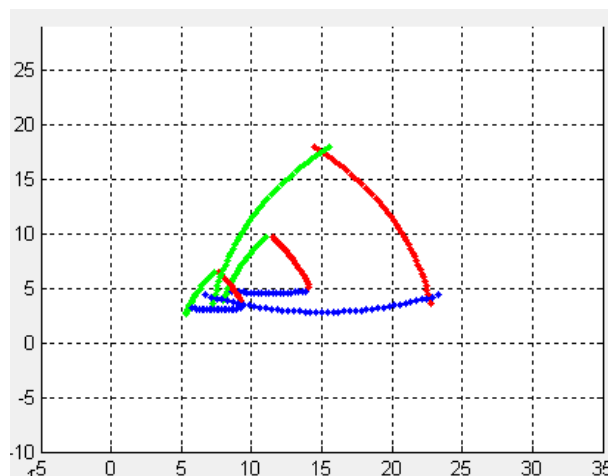
Como se puede observar en la imagen anterior, el área de trabajo de la izquierda corresponde a la misma de la (Figura 67) la de la derecha corresponde con el área de trabajo calculada para los eslabones de 7.5 cm. Según esta figura el área de trabajo que posee el prototipo usando eslabones de 5 cm es más pequeña en comparación a la de los eslabones de 7.5 cm, sin embargo este cambio no es muy significativo. Si se compara el cambio entre las longitudes que es de 50% se puede afirmar que el aumento del espacio de trabajo es un poco mayor, por lo que la proporción de aumento entre el segundo espacio de trabajo es de más de 2:1. La posición (Figura 70) del robot para alcanzar las coordenadas introducidas previamente fue:

Figura 70. Prototipo con eslabones de 7.5 cm en configuración 1



Por último se realizaron las simulaciones de movimiento con eslabones de 10 cm de longitud los cuales son las dimensiones exactas con las que se diseñó y fabricó el prototipo, las coordenadas introducidas para el efector final fueron (15.21,8.98649) y el Angulo \varnothing fue de 0° . El área de trabajo (Figura 71) obtenida fue:

Figura 71. Área de trabajo para eslabones de 5 cm 7.5 cm y 10 cm



Como se puede ver en la imagen anterior, se añadió un espacio de trabajo más que corresponde al área de trabajo si se usaran eslabones de 10 cm de longitud, que corresponden con las dimensiones reales del prototipo que se diseñó, en esta imagen se puede ver claramente la diferencia entre todas, de lo que se puede concluir que el espacio de trabajo obtenido para el prototipo diseñado es la más eficiente de todas si se compara con la proporción de aumento entre las longitudes, en la (Figura 69) el aumento entre las áreas de eslabones de 5 cm y 7.5cm era de poco más del doble sin embargo, si se compara con la de 10 cm, el área de trabajo pasa a ser más de 6:1.

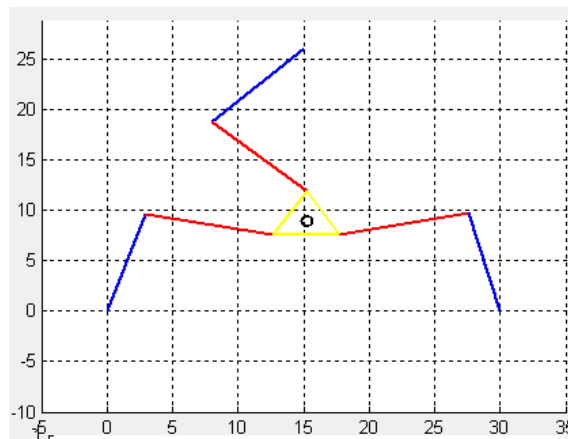
Para simular el ejemplo de las dimensiones del prototipo y de la posición que debe adoptar para que su efector final coincida con el punto que se enunció anteriormente para este caso, las coordenadas articulares en Q y R fueron las coordenadas reales del sistema diseñado:

$$(X_Q, Y_Q) = (30,0)$$

$$(X_R, Y_R) = (15,25.98)$$

La posición (Figura 72) del robot para alcanzar las coordenadas introducidas previamente fue:

Figura 72. Prototipo con eslabones de 10 cm en configuración 1



5.9 SERVOMOTOR DE 180 GRADOS

De acuerdo a la investigación hecha se determina que para alcanzar una coordenada el prototipo puede hacerlo con ocho posibles configuraciones, debido a que cada ángulo θ_1, θ_2 y θ_3 posee dos posibles soluciones dado que sus ecuaciones son de segundo orden, sin embargo solo se programaron 2 de ellas. En la interfaz gráfica, en la sección *Configuración* se puede seleccionar entre las dos. La configuración 1 contiene las la parte positiva de la formula cuadrática de la (Ecuación 4.6).

$$\theta_1 = 2Tan^{-1} \left[\frac{-e_1 + \sqrt{(e_1^2 - e_3^2 + e_2^2)}}{-e_2 + e_3} \right]$$

Los términos $e_1 \dots e_9$ pueden verse en el anexo B. La configuración 2 contiene la parte negativa de la formula cuadrática de la (Ecuación 4.6)

$$\theta_1 = 2Tan^{-1} \left[\frac{-e_1 - \sqrt{(e_1^2 - e_3^2 + e_2^2)}}{-e_2 + e_3} \right]$$

Con cada una de las anteriores configuraciones, se realizaron varias simulaciones para determinar posibles fallas que conllevarían un rediseño del sistema, entre las pruebas realizadas se le asignaron las siguientes coordenadas al efector final:

$$(X, Y) = (12.6903, 8.04054)$$

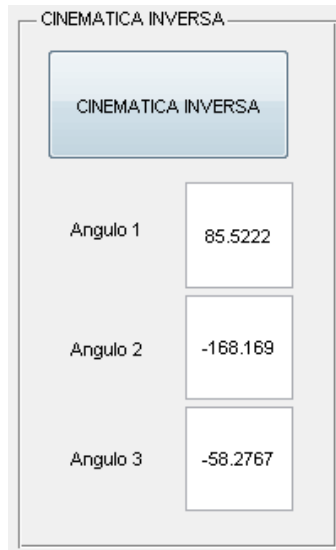
El resultado obtenido (Figura 73) para la configuración 1 fue:

Figura 73. Prototipo en configuración 1



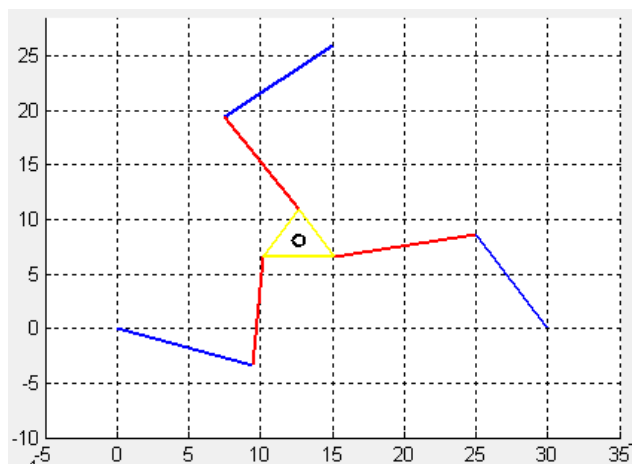
Los ángulos obtenidos (Figura 74) para la anterior posición fueron:

Figura 74. Ángulos Obtenidos primera prueba



De acuerdo a la imagen anterior, se evidencia que el ángulo θ_3 y θ_2 dan resultados mayores a 180° (ángulos complementarios), lo que requería el uso de servomotores diferentes del estándar, es decir servomotores capaces de girar 360° , de tal forma que se procedió probando los mismos parámetros con la segunda configuración (Figura 75):

Figura 75. Prototipo en configuración 2



Los ángulos obtenidos (Figura 76) para la anterior posición fueron:

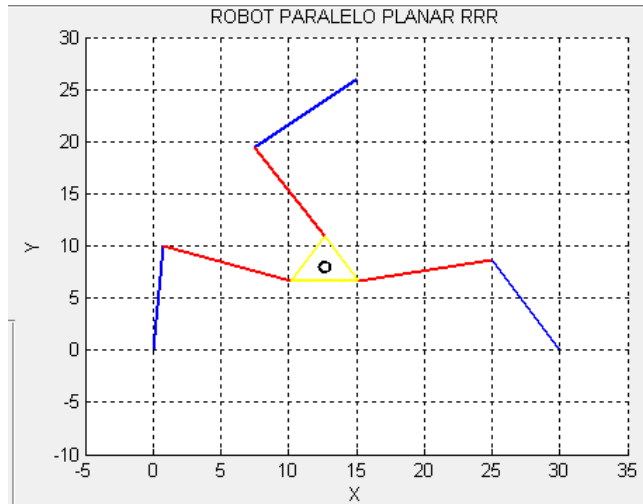
Figura 76. Ángulos obtenidos en la primera prueba con configuración 2



Para esta situación el ángulo θ_3 al igual que en la configuración anterior continuaba dando como resultado un ángulo mayor a 180° , de lo que se concluye que debido a la ubicación de este servomotor con respecto al prototipo, el ángulo que debe formar con respecto al plano que se le asignó siempre va estar entre 180° y 360° . En el caso del ángulo θ_1 dio como resultado un ángulo negativo, valor que el servomotor no puede efectuar, por lo que se recurre al uso de su ángulo complementario, y este también es mayor a 180° .

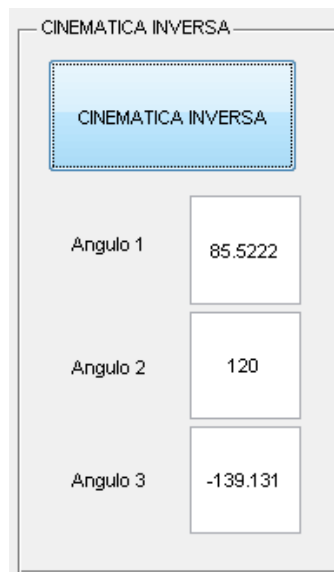
Dadas las condiciones anteriores, en primera instancia se determina que los servomotores a usar en el prototipo deben ser capaces de girar 360° , sin embargo esta decisión trae consigo un aumento significativo en el costo del prototipo de aproximadamente cinco veces más en cada servomotor. Se decide entonces aplicar condicionantes en el código (Figura 77), en los que si los ángulos calculados son mayores a 180° o menores a 0° , se debe usar la ecuación alternativa en ese ángulo:

Figura 77. Prototipo con condicionantes en los ángulos



Los ángulos obtenidos (Figura 78) para la anterior posición fueron:

Figura 78. Ángulos Obtenidos con condicionantes



Gracias a la solución anterior el software nunca genera en las variables θ_1 y θ_2 ángulos mayores a 180° , por lo que en estos dos casos se puede reducir costos, aumentar las ganancias y maximizar la eficacia del prototipo usando servomotores estándar con un ángulo de rotación de $0-180^\circ$. Para el caso del ángulo θ_3 es imposible matemáticamente usando las ecuaciones de la cinemática inversa halladas, que proporcione un ángulo menor a 180° , porque es imposible físicamente que esto pueda suceder, de tal forma que para solucionar este inconveniente se alteró la ecuación de este ángulo de la siguiente forma:

$$\theta_3 = 180 + \left[2 \tan^{-1} \left[\frac{-e_7 + \sqrt{(e_7^2 - e_9^2 + e_8^2)}}{-e_8 + e_9} \right] \right]$$

Es decir que la posición inicial para la ecuación de este ángulo es de 180°, por lo que el servomotor debe rotar el ángulo faltante para completar el ángulo necesario que requiere el sistema para funcionar correctamente, de tal forma que el servomotor que se usa en este ángulo puede ser también un servomotor estándar con un ángulo de giro de 0-180°.

En el diseño de mecanismos siempre es importante usar los recursos que se tienen y no disponer de más a menos que sea absolutamente necesario, debido a que esto genera un ahorro en los costos de fabricación y producción. En el prototipo se logró por medio de razonamiento y análisis reducir el costo en más de 5 veces en componentes vitales como lo son los servomotores, lo que permitió ahorrar dinero y optimizar el diseño, efectos que son importantes en el diseño mecatrónico.

5.10 INTERFAZ DE USUARIO

Con el fin de analizar el comportamiento del prototipo se realizaron varias pruebas del movimiento del efector final (Punto central del triángulo equilátero, punto negro). En primera instancia se introdujeron los valores deseados de las longitudes de los eslabones del prototipo, como se enunció en el documento, físicamente cada eslabón se diseñó de 10 cm razón por la cual en la (Figura 79) , se introdujeron estos valores.

Figura 79. Ingresando longitudes

The image shows a software interface titled "ROBOT PLANAR PARALELO 3RRR". It contains several input fields and a configuration section. The "Eslabones a" and "Eslabones b" fields are both set to the value "10". To the right, there are empty input fields for "Coordenada X" and "Coordenada Y". Below these, a "CONFIGURACION" section contains two radio buttons: "PRIMERA" (which is selected with a blue dot) and "SEGUNDA". A small number "1" is visible below the "PRIMERA" button. To the right of the configuration section, there is a "Phi" field with the value "0".

Usando el botón *Área de trabajo* (Figura 80), podemos determinar el área en el que el efector final se puede mover y con el área calculada y graficada, podemos seleccionar por medio del cursor una coordenada (Figura 81) dentro del eje de graficación del área de trabajo, cabe recordar que si se selecciona una coordenada por fuera de su campo de acción, el robot emitirá un mensaje de error en la parte inferior de la interfaz.

Figura 80. Área de trabajo

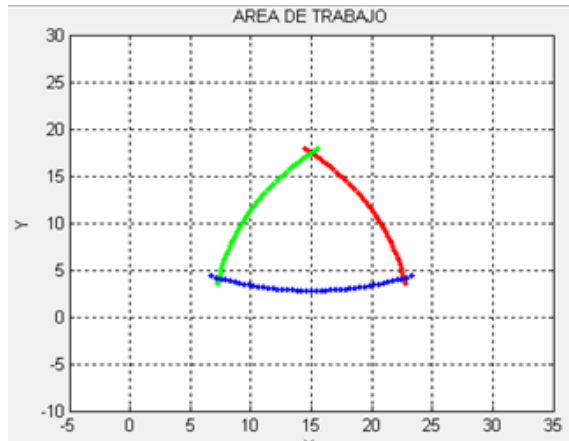


Figura 81. Coordenadas para simulación

Coordenada X	15.105
Coordenada Y	8.71622

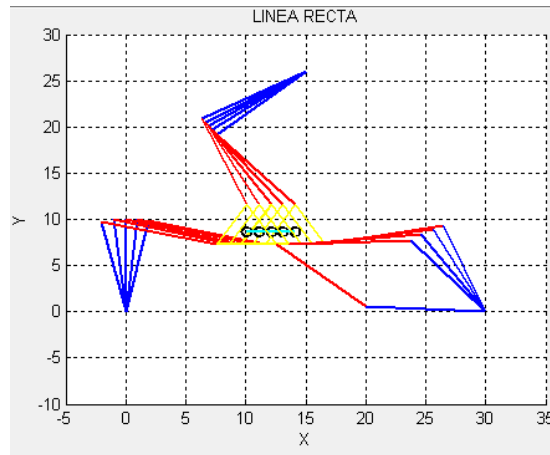
Con el botón de *cinematica Inversa* se ejecuta la simulación del movimiento y posición de cada una de las cadenas cinemáticas del prototipo para llegar a las coordenadas anteriormente establecidas, cada ángulo posee dos soluciones debido a que la ecuación que define su valor es una ecuación de segundo orden y su solución se da por medio de la ecuación cuadrática, es decir que el prototipo tiene dos posibles configuraciones para cada ángulo lo que da 8 posibles configuraciones de ángulos para el movimiento del robot .

Como se puede evidenciar en las imágenes anteriores, el área de trabajo es muy pequeña en comparación con el tamaño del prototipo, se puede notar que el robot es preciso en la ubicación que se definió, y es estable sin embargo el área de trabajo limita mucho su funcionalidad, esto es una característica que define el comportamiento de los robot paralelos, sean planares o espaciales.

Para determinar de forma detallada el comportamiento de los robot planares, con una configuración paralela, es decir con su cadena cinemática cerrada, calculando la precisión y alcance del prototipo se decidió crear rutinas por medio de las cuales se pueda variar el movimiento del prototipo, una de ellas es el botón *línea recta pasos*, gracias a este botón podemos ejecutar una línea (Figura 82) cuyas coordenadas son constantes en el eje Y, tomada de la coordenada en Y definida por el usuario en la sección Coordenadas Para la programación de este proyecto solo se implementaron dos de las ocho posibles configuraciones del sistema, en este botón se muestra paso a paso las

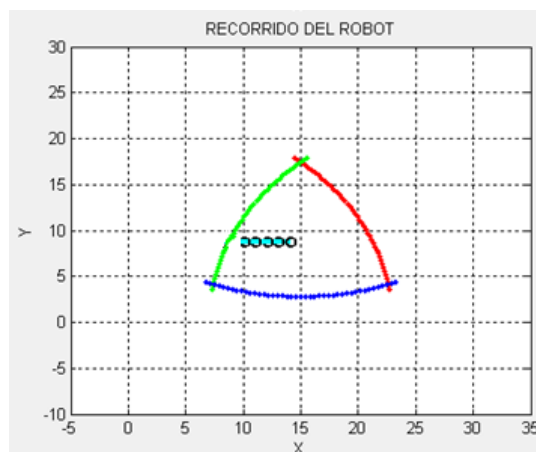
posiciones que el robot debe adoptar para llegar al punto que el usuario solicitó:

Figura 82. Línea Recta con 5 pasos



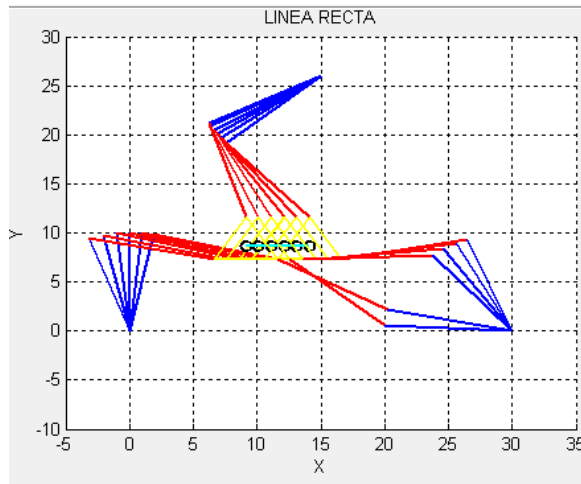
En la imagen anterior se puede observar la secuencia del movimiento del prototipo requerida en su primera configuración para realizar una línea recta, de características descritas anteriormente. En la línea recta de la (Figura 83) se le solicitó al software que hiciera una línea recta tomando como punto inicial las coordenadas introducidas en la interfaz gráfica, y se moviera desde ese punto en pasos de -1 en X, para realizar el movimiento de derecha a izquierda. La trayectoria del robot comparada con el área de trabajo es:

Figura 83. Trayectoria en el área de trabajo de la línea recta de 5 pasos



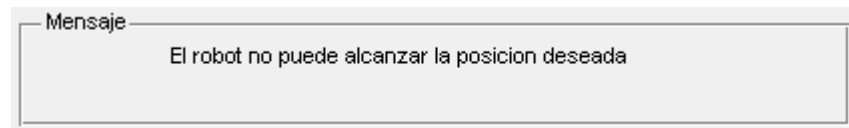
Con las mismas coordenadas iniciales, pero esta vez con un límite de 7 pasos hacia la izquierda, la respuesta (Figura 84) fue de la siguiente forma:

Figura 84. Línea recta de 7 pasos



Para este caso el prototipo solo hizo 6 pasos de los 7 que debió realizar como se puede observar en la figura anterior, la primera hipótesis sugiere que esto sucedió porque el siguiente paso que correspondía al séptimo, era imposible físicamente para el prototipo alcanzarlo, debido a que estaba por fuera de su área de trabajo. El software evalúa la posibilidad de alcanzar un punto antes de que el robot lo haga, con el fin de proteger los componentes del sistema. Debido al error que presentó la solicitud del usuario, el sistema generó el siguiente mensaje (Figura 85) de alerta:

Figura 85. Mensaje de alerta



Se puede suponer que existe un punto en el que la simulación se detiene y arroja el mensaje anterior, esto quiere decir que alguna de las raíces de los ángulos de las articulaciones activas del prototipo, dio imaginaria lo que significa que físicamente no puede alcanzar dicha posición, debido a sus limitaciones mecánicas, es decir la longitud de sus eslabones.

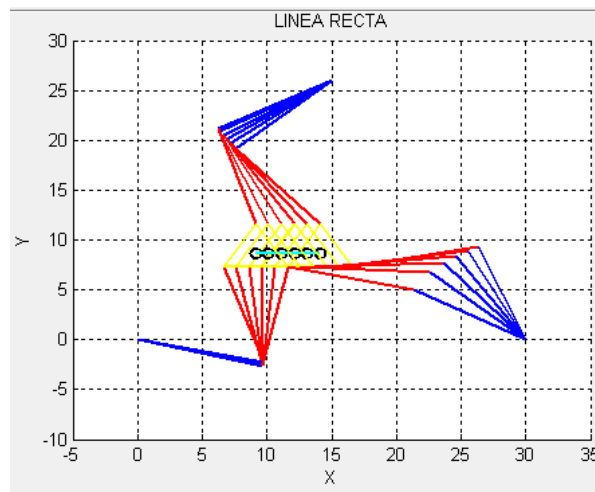
La trayectoria que el robot alcanzó a ejecutar, se puede observar en la siguiente (Figura 86):

Figura 86. Trayectoria en el área de trabajo de la línea recta de 7 pasos



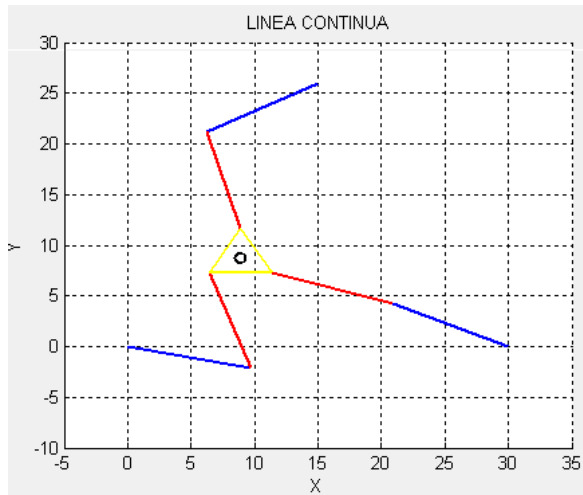
Se realizó la misma prueba con configuración 2, manteniendo la inclinación de la plataforma móvil en 0° (Figura 87) y el resultado obtenido fue el mismo:

Figura 87. Línea recta de 7 pasos en configuración 2



De lo que se concluye y confirma, que la imposibilidad del prototipo para llegar al séptimo punto se debe a que dicho punto está por fuera de su campo de acción y no está relacionado con las posibles configuraciones del robot. Para finalizar el análisis se le solicita en la interfaz que ejecute la misma línea recta de 7 pasos pero esta vez de forma continua en la sección 6, cuando el robot está a un paso de llegar al último, el programa arroja el mensaje de error y la gráfica obtenida refleja la posición real de sus brazos

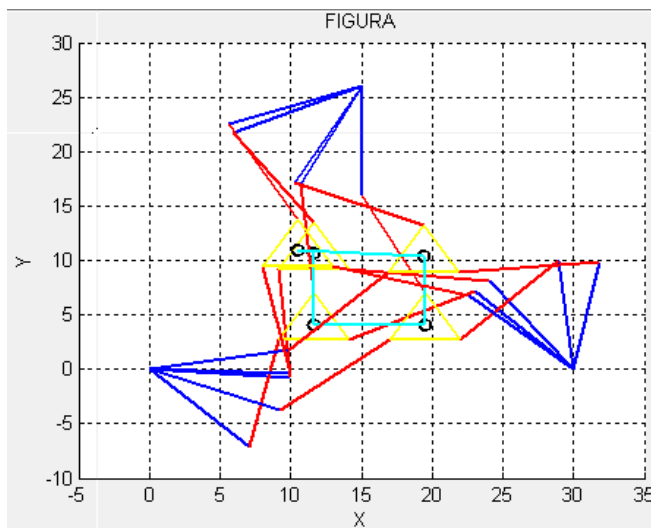
Figura 88. Posición Límite de la línea recta



La Figura 88 sirve como prueba definitiva para asegurar que el robot no puede llegar a la última coordenada debido a que su brazo derecho se encuentra casi totalmente extendido, por lo que dicha coordenada no puede ser alcanzada a menos que aumente el área de trabajo del sistema, y como ya se ha explicado con anterioridad, la única forma de hacerlo es aumentando la longitud de sus eslabones.

En el caso del botón figura también se hace importante conocer el alcance del sistema, en este caso se introdujeron 5 puntos al azar tomados del área de trabajo del robot formando un rectángulo seleccionando la primera configuración, la respuesta del sistema fue la siguiente:

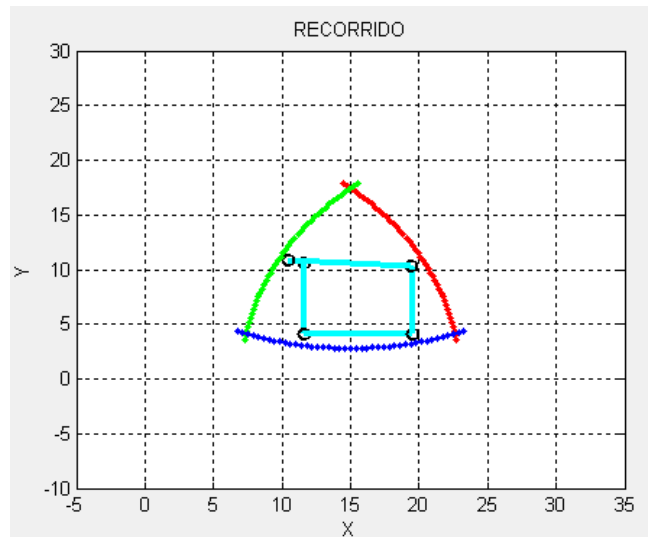
Figura 89. Fin de la figura rectángulo



La simulación (Figura 89)permite determinar las respectivas posiciones que deben adoptar los miembros del robot, en azul se visualizan los eslabones a_1 , a_2 y a_3 en rojo los eslabones b_1 , b_2 y b_3 , en amarillo se identifica la base

móvil, el círculo negro corresponde al efector final del sistema y por último en azul celeste el recorrido del robot. Esta simulación tiene un tiempo de 2 segundos entre posiciones del robot, por lo que visualizar en tiempo real y fácilmente el comportamiento del robot

Figura 90. Trayectoria del robot



En la Figura 90 se observa claramente el recorrido del robot, en el que se detalla que se encuentra dentro de su campo de acción, aunque su último punto está muy cerca del límite, si alguno de los puntos de la figura introducida se encuentra por fuera del área de trabajo, el sistema arroja el mensaje de error y se detiene la ejecución del dibujo.

5.11 PRESICION DEL ROBOT

Inicialmente se diseñó que el grado de precisión fuera de un 85% ya que es un prototipo, las razones válidas de que no sea superior son:

- Como es un prototipo se optó por no utilizar unos servomotores de alta calidad, ya que son muy costosos, en su lugar se utilizaron unos de menor calidad, esto quiere decir que tienen menor precisión, y sus engranajes son de plástico por lo tanto tendrán mayor desgaste por uso.
- Debido a que los pasadores son tornillos tienden a soltarse con varios movimientos de los eslabones, esto quiere decir que llegará un punto donde uno o más eslabones estará menos firme por lo tanto el robot hará una posición completamente diferente a la indicada en la interfaz de usuario, lo recomendable es ajustar los tornillos periódicamente e intentar que sean ajustados con la misma fuerza.

A continuación se da a conocer una (Tabla 4) la cual nos mostrara el porcentaje de error, frente a cada 0.5 centímetro cuadrado.

Tabla 4. Porcentaje de precisión

Porcentaje (%)	Área de error
100%	0.5 cm^2
95%	1.0 cm^2
90%	1.5 cm^2
85%	2.0 cm^2
80%	2.5 cm^2
75%	3.0 cm^2

La idea es cumplir con el diseño es decir que el robot no supere el 2.0 cm^2 , si este error se supera, quiere decir que están mal ajustados los eslabones, o que tal vez el origen del primer eslabón frente al ángulo del servomotor no es la adecuada, en la Figura 91 se muestra la posición deseada por el usuario, y en la Figura 92 se evidencia el lugar donde el robot se posicionó.

Figura 91. Posición deseada en la interfaz

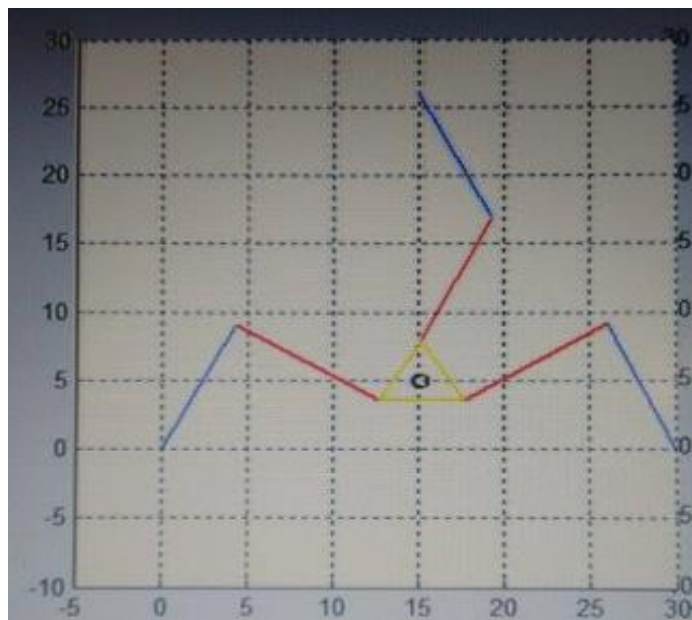
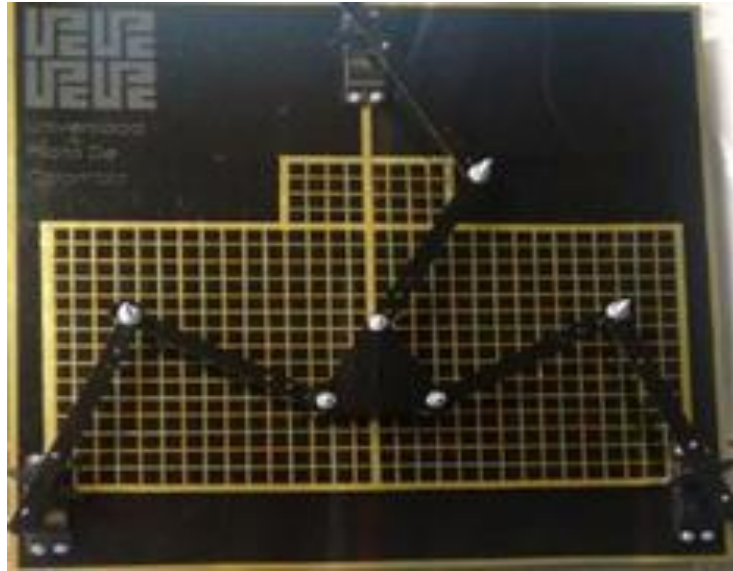


Figura 92. Posición real del robot



Con una longitud de los eslabones de 10 cm, se aplicó una coordenada de (15,5) y el robot realizó la posición de (15, 5.5), esto quiere decir que está en el límite del error pero aún sigue en el rango, por lo cual cumple con el diseño.

5.12 ANÁLISIS DE COSTOS

Es importante analizar la eficiencia del prototipo en todos los aspectos, En este caso se ha concentrado el análisis en la parte mecánica, electrónica y de programación del sistema sin embargo, por más efectivo y preciso que sea un producto debe ser considerado el costo de su producción, fabricación, ensamblaje y horas de trabajo con el fin de determinar la aceptación que tendrá el producto y cuál sería el tipo de consumidor que tendría mayor probabilidad de adquirirlo. En todos los sectores del mercado en los que se trate de incursionar hoy en día , por lo general hay al menos un competidor con una tecnología o aplicación similar a la que se intente desarrollar debido a la densidad poblacional a la que nos encontramos actualmente y que seguirá aumentando en el futuro de forma exponencial, por tal razón a parte de superar la competencia en calidad y servicio es supremamente importante hacerlo en la relación costo-beneficio, es decir, no es necesario que el producto que se desarrolle sea más económico en comparación a la competencia, si no que el valor que el consumidor deba pagar por adquirirlo represente una inversión significativa la cual podrá ser retribuida en menor cantidad de tiempo o cuyas ganancias sean significativamente mayores que las que podría producir el producto de la competencia.

El primer paso para orientar el producto es prioritario conocer lo que cuesta producirlo, para ello se han dividido o separado sus costos en función de la etapa del proceso requerida para generar un prototipo terminado, los cuales son: diseño, fabricación y ensamble.

5.12.1 DISEÑO

Esta etapa fue la más larga de todas en el momento en que el prototipo se encontraba en el inicio del proceso de investigación y experimentación, sin embargo con un primer prototipo terminado y completamente funcional, se considera que el tiempo que tome para una persona técnicamente capacitada con las herramientas y conocimientos adquiridos del proceso experimental sea mucho menor por lo que en este aspecto el valor final del producto podría tener una reducción si fuese necesario, sin embargo el presente calculo (Tabla 5) toma en cuenta el tiempo que tomo su diseño en la parte experimental debido a que el desarrollo del sistema es la parte esencial del producto, la tecnología y herramientas que se usaron para generar el producto en esta primera etapa:

Tabla 5. Costos de diseño en parte experimental

DISEÑO				
ETAPA	CANT. HORA	COSTO HORA	COSTO TOTAL	
Diseño mecánico	40	\$ 3.500,00	\$ 140.000,00	
Diseño electrónico	30	\$ 2.000,00	\$ 60.000,00	
Programación	80	\$ 5.000,00	\$ 400.000,00	
SOFTWARE				
PROGRAMA	COSTO \$US	COSTO \$COP	TIEMPO	COSTO/ MES
CAD	\$ 1.500,00	\$ 4.552.350,00	12 Meses	\$ 379.362,50
MATLAB R2013B	\$ 2.650,00	\$ 8.042.485,00	12 Meses	\$ 670.207,08

En la industria actual los proyectos que se realizan por lo general son diferentes a pesar de que sea el mismo producto, esto se debe a las especificaciones particulares de los consumidores los cuales determinan según sus necesidades, que es lo que requieren y es deber del productor ofrecer una solución, de tal manera que es muy común que tengan que haber rediseños en todas las áreas del producto en función de satisfacer las variadas necesidades de los clientes, por consecuente se establecen por proyectos unas horas estimadas límite para diseñar cada producto.

En esta etapa se dividió la mano de obra por etapas del diseño, que están compuestas por diseño mecánico, electrónico y la programación del producto, las horas por etapa se pueden observar en la tabla anterior y estas horas están valoradas según los ingresos mensuales de un profesional en cada una de estas áreas.

Como se puede observar en la tabla anterior se tuvo en cuenta el costo de las licencias Premium de los software que se requieren para completar la etapa de diseño, estos software tienen una licencia que dura 12 meses el valor de estas licencias está en dólares, razón por la cual en la columna siguiente aparece su valor en pesos colombianos usando la TRM del día 05 de octubre del 2015.

Para cuestiones de costeo se decidió dividir el valor total de las licencias en los 12 meses que estaría activa y este valor se le sumara a cada uno de los proyectos que se diseñen.

De lo anterior se concluye que:

Tabla 6, Costos por mes

ETAPA	COSTO TOTAL
Diseño mecánico	\$ 140.000,00
Diseño electrónico	\$ 60.000,00
Programación	\$ 400.000,00
Licencia software CAD	\$ 379.362,50
Licencia MATLAB	\$ 670.207,08
Costo Diseño	\$ 1.649.569,58

El costo en la etapa de diseño teniendo en cuenta mano de obra y software requeridos para finalizar este paso es de \$ 1.649.569,58, cabe aclarar que este costo puede variar en función de la TRM de diseño del producto.

5.12.2 Fabricación

Con la etapa de diseño completa, se obtienen los planos mecánicos respectivos para continuar con la etapa de fabricación en la cual se utiliza principalmente el proceso de corte por láser, debido a que el material que se usó para este prototipo es el acrílico. Para cortar un material de estas características se le asigna a la maquina láser una potencia y velocidad determinadas para ese material en particular, por lo que el trabajo se cobra por unidad de tiempo, pero hay materiales en los que ya se estandarizó el valor de corte por metro lineal

Para computar los metros lineales de corte hay que calcular los contornos de las piezas a cortar. Por ejemplo un cuadrado de 10 cm de lado va a tener 0,4 m de corte, si se tiene que cortar 10 cuadrados iguales serian entonces 4 m, a este valor se le multiplica por el precio por ml de corte en el material que se necesita cortar y se obtiene el precio aproximado del trabajo. Es importante aclarar que esto no incluye el costo del material, a continuación se puede evidenciar los costos de fabricación para el prototipo

Tabla 7. Costos del corte del acrílico

CORTE LASER						
PIEZA	MATERIAL	ESPESOR (mm)	CANTIDAD	CONTORNO(m)	COSTO UND(\$/m)	TOTAL
Base con Logo UPC	Acrílico	3	1	1,36	\$ 3.050,00	\$ 4.148,00
Orificios de la base (servomotores)	Acrílico	3	1	0,36	\$ 3.050,00	\$ 1.098,00
Grabado logo Universidad	Acrílico	3	1	1	\$ 14.650,00	\$ 14.650,00
Base sobre base	Acrílico	3	1	1,336	\$ 3.050,00	\$ 4.074,80
Eslabón A	Acrílico	3	3	0,29	\$ 3.050,00	\$ 2.653,50
Eslabón B	Acrílico	3	3	0,26	\$ 3.050,00	\$ 2.379,00
Efactor Final	Acrílico	3	1	0,201	\$ 3.050,00	\$ 613,05
Uniones lados	Acrílico	3	2	0,708	\$ 3.050,00	\$ 4.318,80
Base, frente y atrás	Acrílico	3	3	0,705	\$ 3.050,00	\$ 6.450,75
Curvas	Acrílico	3	12	1	\$ 3.050,00	\$ 36.600,00
					TOTAL	\$ 76.985,90

Las piezas mostradas en la tabla anterior se pueden encontrar en el anexo F. De acuerdo a lo anterior se puede concluir que entre menos cortes tengan las piezas más económico será el proceso de corte, y adicionalmente agregar un grabado sobre la placa de acrílico representa un costo adicional que puede ser minimizado con el fin de obtener mayor ganancia. Incluyendo el costo del material, el valor de esta etapa asciende a \$96.985

5.12.3 Ensamble

Para la etapa de ensamble se dispone de un presupuesto menor gracias a que los elementos de sujeción del proyecto son unos pocos tornillos, arandelas y tuercas, así mismo el tiempo que requiere el ensamble de las partes que componen el prototipo en su mayoría son muy simples de armar, por lo que esta parte del proceso no se le asigna costo alguno, sin embargo hay que tener en cuenta el costo de los elementos principales del proyecto en esta etapa que son:

Tabla 8. Costo de elementos principales

ELEMENTOS PRINCIPALES			
ELEMENTO	CANTIDAD	COSTO UND	COSTO TOTAL
Servomotor 0-180°	3	\$ 38.000,00	\$ 114.000,00
Arduino UNO R3	1	\$ 69.000,00	\$ 69.000,00
Jumpers	15	\$ 500,00	\$ 7.500,00
		TOTAL	\$ 190.500,00

Finalmente realizando una sumatoria de todos los costos anteriores y de la ganancia que se esperaba recibir, se obtiene la siguiente tabla:

Tabla 9. Costo total

COSTO TOTAL	
ETAPA	COSTO
Diseño	\$ 1.649.569,58
Producción	\$ 76.985,90
Ensamble	\$ 0,00
Materiales principales	\$ 190.500,00
Ganancia Esperada	100%
TOTAL	\$ 3.834.110,96

Con una ganancia del 100% del costo total del prototipo, incluyendo software licenciado por un año, los materiales principales del prototipo como el acrílico, servomotores, arduino, entre otros y adicionando todos los costos de fabricación, se obtiene que el valor de venta de un prototipo con características similares (tamaño, alcance, operatividad) al que se desarrolló en el presente documento es de aproximadamente \$ 3.834.150 COP

5.12.4 Mercado actual

Según la investigación realizada al mercado actual no existe una maquina con las características que tiene en este punto el prototipo diseñado, se han encontrado algunos mecanismos similares pero más primitivos y en muchos otros casos son más avanzados (tecnológicamente hablando), sin embargo, se considera prudente realizar una comparativa con los dos casos descritos debido a que esto ayudara a ubicar el robot planar en el mercado, permitiendo determinar el mejor sector en el que se puede desenvolver y por último, asignar de forma precisa un buen precio que algún consumidor podría estar dispuesto a pagar en función de sus necesidades, las soluciones que se le pueden brindar y la competencia actual.

De acuerdo a la información provista en los capítulos anteriores, una de las principales aplicaciones del prototipo que se ha diseñado es la de un pantógrafo en automático y digital, esto es debido a sus características las cuales le permiten realizar los dibujos que el usuario introduzca por medio de la interfaz gráfica de MATLAB, en esta área podemos encontrar diferentes productos que el mercado ofrece y que tienen bastantes similitudes con el robot planar que se ha desarrollado por lo cual se pueden considerar como competencia directa, entre ellos encontramos:

5.12.5 Pantógrafos Estándar

Figura 93. Sistema de rieles para bombillos



Hoy en día existen sistemas netamente mecánicos como los que se pueden observar en la imagen anterior que son totalmente manuales y permiten el desplazamiento de objetos por diferentes zonas, el sistema de la (Figura 93) es un sistema que permite variar la ubicación de bombillos en un apartamento por medio de los rieles que posee, este sistema puede variar tanto ubicación en el plano XY como la altura Z. una de las más notorias desventajas de este tipo de sistemas es variar la ubicación de sus elementos en lugares con un techo alto en comparación al suelo, como es manual se deberían usar implementos que funcionen como una extensión del cuerpo para variar la posición de los bombillos mientras que si fuese automático como lo es el robot planar, bastaría con tener una periferia en algún lugar de la habitación que sea accesible para todas las personas de la casa o se podría implementar un sistema de movimiento inalámbrico del prototipo. El robot planar puede tener una aplicación similar, si se le agrega un sistema que transforme el voltaje alterno que puede proporcionar una toma corriente en voltaje directo (debido a que el pantógrafo manual, usa 24 VDC para alimentar sus bombillos), este sistema en teoría no debería aumentar su precio más de un 20%, por lo que podría elevarse el costo a \$ 4.600.980,00 COP. El valor del pantógrafo manual asciende a \$ 2.900.000,00 COP, si se comparan netamente los dos valores el robot planar RRR es más costoso sin embargo la relación costo-beneficio de la que se hablaba anteriormente es sin duda mucho mayor, ya que por solo un 31 % más del precio del pantógrafo manual, el consumidor puede obtener un pantógrafo automático, que puede ser multivoltaje por lo que podría usar cualquier tipo de bombillos y que puede cambiar su posición a distancia lo que reduce el desgaste físico y las posibilidades de sufrir accidentes, sin contar que el prototipo robot planar RRR cuenta con una ganancia estimada del 100%, que puede ser reducida para así disminuir su precio más.

Es importante aclarar que por más eficiente, efectiva y económica (hablando de relación costo-beneficio) siempre va existir el factor monetario en los consumidores que en muchos casos toman la decisión de adquirir productos en

función de su capacidad de endeudamiento, por lo que en dichos casos la calidad del producto pasa a un segundo plano.

6 TRABAJOS FUTUROS

El robot paralelo planar de tres articulaciones rotacionales es un proyecto muy ambicioso en el cual se pueden visualizar muchas aplicaciones en diferentes sectores, todas ellas son innovadoras, eficientes y útiles. Entre las principales ventajas que este prototipo tiene, se puede destacar su versatilidad, pensar en el hecho que el prototipo pueda trabajar en el sector industrial no es algo alejado de la realidad, pero esto requiere más tiempo de desarrollo e investigación, si se le añadiera una etapa de potencia al sistema en el que en vez de tener como efector final un lápiz que realice dibujos sobre el papel, se cambiara por un rayo láser que sea capaz de cortar materiales como acrílico, metales, entre otros, convertiría al prototipo en una maquina cortadora capaz de realizar cualquier tipo de corte dentro de su espacio de trabajo, si se usa el mismo principio de movimiento que se le programó pero se le agregara un set de herramientas reemplazables en el lugar de sujeción de su efector final, tales como buriles de corte o grabado y adicionalmente se controlara la velocidad de giro de la herramienta para corte o grabado, este sistema sería capaz de realizar tareas de grabado de Placas, señalización, marcación entre muchas otras. A lo largo del diseño y fabricación del sistema se pensaron ideas que podrían ser útiles para un prototipo como este, sin embargo dichas ideas o funciones estaban por fuera de los objetivos que el proyecto tenía que satisfacer, como una forma de ingreso de datos más eficiente y con un campo muy grande de aplicaciones, el incluir la tecnología de visión artificial como sistema de ingreso de coordenadas o de patrones para que el robot lo replique puede ser una excelente opción, esta función sería capaz de reconocer por medio del tratamiento de imágenes, cualquier figura que el usuario desee replicar como por ejemplo una maquina capaz de hacer retratos de cualquier persona, no obstante si se combinaran algunas de las ideas anteriores se podría cubrir un campo mucho más amplio de operatividad del sistema, como por ejemplo si la misma máquina que gracias a la visión artificial es capaz de replicar cualquier figura o retrato que desee, fuera capaz de replicar la imagen sobre cualquier superficie, es decir realizar un grabado de dicha figura en una placa de metal usando herramientas diferentes para dicha tarea, esta sería sin lugar a dudas una maquina capaz de satisfacer las necesidades de muchos sectores de la economía.

7 CONCLUSIONES

- Las ecuaciones cinemáticas directas permiten calcular las coordenadas y la orientación del efector final en función de las longitudes de los eslabones y de los ángulos que estos forman en sus tres articulaciones rotacionales, se puede afirmar entonces que las ecuaciones de cinemática inversa halladas en este documento, permiten calcular los ángulos formados por los eslabones en las articulaciones del prototipo en función de las coordenadas del efector final introducidas por usuario y de las longitudes de los eslabones.
- Tanto las ecuaciones de cinemática directa como inversa se hallaron de forma parametrizada, en términos de las variables requeridas para el cálculo de posición y orientación en el caso de directa y de los ángulos en las articulaciones en el caso de la inversa. La parametrización de las ecuaciones permitió un rápido análisis y comparación del comportamiento del prototipo si las variables requeridas para el cálculo varían, como por ejemplo en el caso de las longitudes de los eslabones, en las que se pudo evidenciar que el área de trabajo es directamente proporcional a la longitud del eslabón más pequeño del sistema.
- El prototipo se diseñó con un solo punto de sujeción para el efector final que para el caso educativo es un lápiz, el cual gracias a los movimientos del prototipo es capaz de dibujar la figura que el usuario introduzca por medio de la interfaz gráfica programada, sin embargo para optimizar el sistema se podría rediseñar la plataforma móvil añadiendo más orificios de sujeción con el fin de mover el área de trabajo del prototipo en el eje X,Y sin cambiar el diseño de las demás partes del sistema, es importante resaltar que la única opción para aumentar o disminuir el área de trabajo del sistema que se diseñó, es variando las longitudes de los eslabones usando las ecuaciones de la geometría del mecanismo que se usaron en el prototipo.
- A diferencia de los robot de cadena cinemática cerrada ya sean planares o espaciales, los robot de cadena cinemática cerrada tienen una menor área de trabajo, de tal forma que para equiparar el espacio de trabajo el tamaño de los robot seriales, los robot paralelos deben ser mucho más grandes, debido a lo anterior estos robot deben ser empleados en tareas que requieran que su espacio de trabajo se mantenga estable y exista un amplio campo para que puedan actuar libremente sin interferencias.

- El control del robot se efectuó implementando en primera instancia la geometría del mecanismo, con la que se despejaron las ecuaciones de movimiento y de velocidad en cada punto del sistema. El modelamiento matemático del prototipo en cuanto a posición se obtuvo usando las mencionadas ecuaciones de la cinemática del mismo y para el caso de las velocidades articulares y del efector final, se modelaron con la matriz jacobiana desarrollada en el presente documento.
- Los eslabones del robot poseen varios orificios de sujeción debido a que se consideró importante en el diseño del sistema generar un prototipo flexible y amigable con la mayor cantidad de situaciones a las que podría verse enfrentado, gracias a estos orificios se puede variar físicamente la longitud activa de sus eslabones y por consiguiente su espacio de trabajo. Este prototipo puede adaptarse de forma manual para moverse en áreas de trabajo más pequeñas, sin embargo no puede superar el área de trabajo que se puede obtener usando los orificios de los extremos de los eslabones, es decir los 10 cm de longitud por eslabón.
- Por medio de los análisis de esfuerzos presentes bajo condiciones normales de trabajo, se pudo determinar que las dimensiones de los eslabones y de la plataforma móvil son adecuados, y presentan un buen resultado para generar un rendimiento efectivo del sistema, de igual manera se pudo establecer que el material que se escogió en primera instancia como tentativo para la fabricación del prototipo que fue el acrílico presenta unas especificaciones técnicas que garantizan una excelente resistencia para la aplicación diseñada.
- El prototipo que se diseñó, tiene aplicaciones en muchos sectores de la ingeniería, principalmente en el corte de láminas para formar piezas sin embargo esta aplicación aún requiere más desarrollo e investigación, debido a que se debe diseñar una nueva etapa que controle la forma de corte. El robot se creó en una primera instancia para el entorno interactivo y educativo de carreras afines con la robótica, por tal razón el efector final con el que se realizaron las pruebas de campo es un lápiz el cual simplemente tiene como función dibujar lo que el usuario introduzca en la interfaz siempre y cuando se encuentre de su área de trabajo.
- Para el ámbito educativo este prototipo es muy eficiente debido a que integra los conceptos requeridos en el campo de la robótica, su interfaz gráfica permite al usuario variar los parámetros que desee del sistema con un entorno

amigable que permite visualizar el comportamiento del prototipo de forma eficaz, en el ámbito mecánico el sistema que se desarrolló permite al operador determinar cuáles serán las posibles configuraciones en las que el prototipo puede llegar al punto deseado o realizar la figura que se introduzca, finalmente la integración de estos dos sistemas gracias a la comunicación vía puerto serial, permite monitorear en tiempo real el funcionamiento real del robot comparándolo con el que se puede visualizar en la interfaz desarrollada.

8 BIBLIOGRAFÍA

- BARRIENTOS, Antonio, et al. Fundamentos De Robótica. Editado por Concepción Fernández Madrid. 2 ed. España. McGraw-Hill, 1997.327p. ISBN 84-481-0815-9.
- LLANOS, Fátima. Lliguin,Angel. Control Electrónico por comandos de voz para un robot zoomórfico tipo mascota interactiva. Tesis de grado para ingeniería electrónica y Computación. Riobamba, Ecuador: Escuela superior politécnica de Chimborazo. Facultad de informática y electrónica.2010. p.164.
- ARACIL,Rafael.et.al` . ROBOTS PARALELOS: Maquinas con un pasado para una robótica del futuro.Enero, 2006. Vol 3. no.1.ISSN 1697-7912. P.16-28.
- VIVAS,Andres. robótica paralela: aplicaciones industriales, modelado y control En: Introducción. *Universidad del Cauca, Departamento de Electrónica, Instrumentación y Control, Popayán, Colombia* .2006.p.8
- MARTINEZ,Noe.ROJAS,Angel reproducción de trayectorias en un robot paralelo 3rrr configurable. Coyoacán, México, D.F. 19 -21 DE SEPTIEMBRE, 2012. ISBN 978-607-95309-6-9.p 445-451.
- BRIONES, Juan. Diseño, análisis y construcción de un robot paralelo traslacional.Tesis para obtener el grado de maestro en tecnología avanzada. Instituto politécnico nacional Queretaro.2009. p. 68.
- CORREA, Alexander. Sistemas robóticos teleoperados, Teleoperated Robotics Systems En: Ciencia e ingeniería neogranadina. noviembre, 2005. no. 15, p. 62-72.
- Aracil,Rafael.ARMADA,Manuel.BALAGUER,Carlos, Robots de servicio. Madrid.2008.vol 5. no 2. ISSN 1697-7912.p. 6-13.
- YAÑEZ,Ricardo. Resolución de mecanismo paralelo Planar 3RRR impulsado por actuadores eléctricos.Tesis para obtener el grado de maestro en tecnología avanzada Queretaro. noviembre.2007.p.118.
- Tsai L.W, *Robot Analisis. The mechanics of serial and parallel manipulators*. EUA: Wiley Interscience, 1999.
- Williams II R. L. and Shelley B. H. Inverse Kinematics for planar parallelmanipulators. American Society Of Mechanical Engineers, 1997.
- Proyecto Arduino ¿Qué es arduino?, [en línea] <<https://proyectoarduino.wordpress.com/%C2%BFque-es-arduino/>> [citado en 24 de agosto de 2015]
- Sastre,Toni Ruiz ELECTROINSAIMADA TUTORIALES RASPBERRY PI , ARDUINO ,ELECTRONICA Y MUCHO MAS. Tutorial Arduino : Servomotor, <<http://www.electroensaimada.com/servomotor.html>> [citado en 24 agosto de 2015]

- ARDUINO. ARDUINO /GENUINO UNO <<https://www.arduino.cc/en/Main/ArduinoBoardUno>>[citado en 26 de agosto de 2015]
- Arduino y Matlab Tutorial 01 Como Configurar arduino y matlab para realizar... IngJulianVelasco. <<http://www.youtube.com/watch?v=TitVKsC9tX0>> [citado en 18 de julio de 2015]
- Gil,Jorca. Robótica e inteligencia artificial, Componentes/elementos de un sistema robot [en línea] <http://www-pagines.fib.upc.es/~rob/protegit/treballs/Q2_03-04/rob_ia2/elem1.htm> [citado en 15 de septiembre de 2015]

ANEXO A

Geometría del mecanismo:

Como se puede ver en la (Figura 97), se estableció un plano coordenado en el cual el punto P que es el primer servomotor del prototipo tiene como coordenadas (0,0), el punto Q (30,0) y el punto R (15,25.98). Con base a lo anterior se definieron los ángulos activos, que son aquellos sobre los que tenemos control θ_1, θ_2 y θ_3 y los pasivos, que son aquellos sobre los que no se posee control a_1, a_2 y a_3

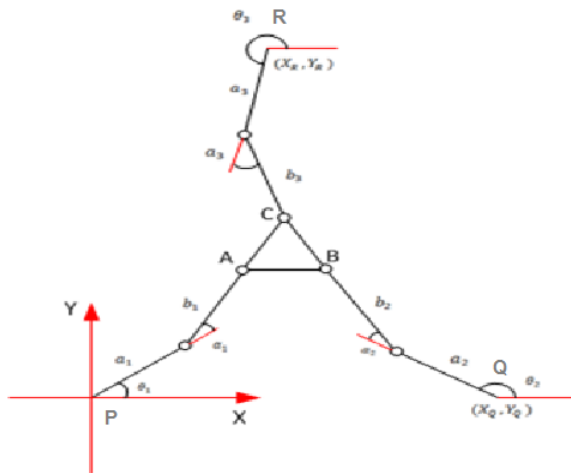


Figura 94. Plano coordenado

Haciendo énfasis en el triángulo equilátero (Figura 98) que une las 3 cadenas cinemáticas:

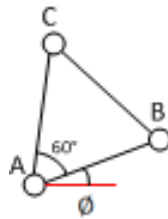


Figura 95. Triangulo

$$(A.1) \quad X_B = X_A + h \cos(\phi)$$

$$(A.2) \quad Y_B = Y_A + h \sin(\phi)$$

$$(A.3) \quad X_C = X_A + h \cos\left(\phi + \frac{\pi}{3}\right)$$

$$(A.4) \quad Y_C = Y_A + h \sin\left(\phi + \frac{\pi}{3}\right)$$

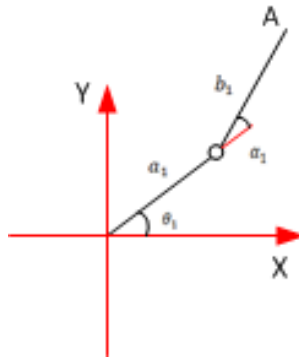


Figura 96. Primera cadena cinemática

De la primera Cadena cinemática (Figura 99) se tiene que:

$$(A.5) \quad X_A = a_1 \cos(\theta_1) + b_1 \cos(\theta_1 + \alpha_1)$$

$$(A.6) \quad Y_A = a_1 \sin(\theta_1) + b_1 \sin(\theta_1 + \alpha_1)$$

Agrupando términos

$$X_A - a_1 \cos(\theta_1) = b_1 \cos(\theta_1 + \alpha_1)$$

$$Y_A - a_1 \sin(\theta_1) = b_1 \sin(\theta_1 + \alpha_1)$$

Elevando al cuadrado ambas ecuaciones

$$X_A^2 - 2X_A a_1 \cos(\theta_1) + a_1^2 \cos^2(\theta_1) = b_1^2 \cos^2(\theta_1 + \alpha_1)$$

$$Y_A^2 - 2Y_A a_1 \sin(\theta_1) + a_1^2 \sin^2(\theta_1) = b_1^2 \sin^2(\theta_1 + \alpha_1)$$

Sumando las ecuaciones:

$$\begin{aligned} X_A^2 + Y_A^2 - 2X_A a_1 \cos(\theta_1) - 2Y_A a_1 \sin(\theta_1) + a_1^2 \cos^2(\theta_1) + a_1^2 \sin^2(\theta_1) \\ = b_1^2 \cos^2(\theta_1 + \alpha_1) + b_1^2 \sin^2(\theta_1 + \alpha_1) \end{aligned}$$

Haciendo Factor común:

$$\begin{aligned} X_A^2 + Y_A^2 - 2X_A a_1 \cos(\theta_1) - 2Y_A a_1 \sin(\theta_1) + a_1^2 (\cos^2(\theta_1) + \sin^2(\theta_1)) \\ = b_1^2 (\cos^2(\theta_1 + \alpha_1) + \sin^2(\theta_1 + \alpha_1)) \end{aligned}$$

Simplificando con la identidad: $\cos^2(\theta_1) + \sin^2(\theta_1) = 1$

$$(A.7) \quad X_A^2 + Y_A^2 - 2X_A a_1 \cos(\theta_1) - 2Y_A a_1 \sin(\theta_1) + a_1^2 - b_1^2 = 0$$

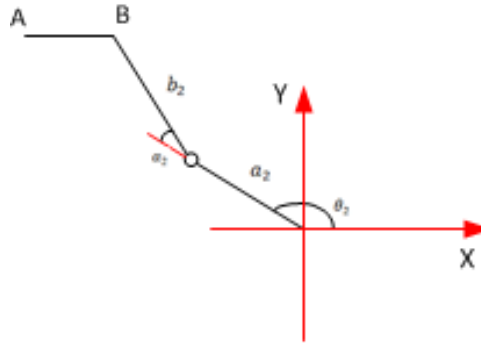


Figura 97. Segunda cadena cinemática

De la segunda Cadena cinemática (Figura 100) se tiene que:

$$(A.8) \quad X_B = X_Q + a_2 \cos(\theta_2) + b_2 \cos(\theta_2 + \alpha_2)$$

$$(A.9) \quad Y_B = Y_Q + a_2 \sin(\theta_2) + b_2 \sin(\theta_2 + \alpha_2)$$

Reemplazando las ecuaciones A.1 y A.2 en las ecuaciones A.8 y A.9:

$$X_A + h \cos(\phi) = X_Q + a_2 \cos(\theta_2) + b_2 \cos(\theta_2 + \alpha_2)$$

$$Y_A + h \sin(\phi) = Y_Q + a_2 \sin(\theta_2) + b_2 \sin(\theta_2 + \alpha_2)$$

Agrupando términos:

$$X_A + h \cos(\phi) - a_2 \cos(\theta_2) - X_Q = b_2 \cos(\theta_2 + \alpha_2)$$

$$Y_A + h \sin(\phi) - a_2 \sin(\theta_2) - Y_Q = b_2 \sin(\theta_2 + \alpha_2)$$

Elevando al cuadrado ambas ecuaciones:

$$\begin{aligned} X_A^2 + h^2 \cos^2(\phi) + a_2^2 \cos^2(\theta_2) + X_Q^2 + 2X_A h \cos(\phi) - 2X_A a_2 \cos(\theta_2) - 2X_A X_Q \\ - 2h a_2 \cos(\phi) \cos(\theta_2) + 2a_2 X_Q \cos(\theta_2) - 2h X_Q \cos(\phi) \\ = b_2^2 \cos^2(\theta_2 + \alpha_2) \end{aligned}$$

$$\begin{aligned} Y_A^2 + h^2 \sin^2(\phi) + a_2^2 \sin^2(\theta_2) + Y_Q^2 + 2Y_A h \sin(\phi) - 2Y_A a_2 \sin(\theta_2) - 2Y_A Y_Q \\ - 2h a_2 \sin(\phi) \sin(\theta_2) + 2a_2 Y_Q \sin(\theta_2) - 2h Y_Q \sin(\phi) \\ = b_2^2 \sin^2(\theta_2 + \alpha_2) \end{aligned}$$

Sumando ambas ecuaciones

$$\begin{aligned}
 & X_A^2 + Y_A^2 - 2X_A X_Q - 2Y_A Y_Q + X_Q^2 + Y_Q^2 + h^2 (\text{Cos}^2(\emptyset) + \text{Sen}^2(\emptyset)) \\
 & + a_2^2 (\text{Cos}^2(\theta_2) + \text{Sen}^2(\theta_2)) + 2X_A h \text{Cos}(\emptyset) + 2Y_A h \text{Sen}(\emptyset) \\
 & - 2X_A a_2 \text{Cos}(\theta_2) - 2Y_A a_2 \text{Sen}(\theta_2) - 2X_A X_Q - 2Y_A Y_Q \\
 & - 2h a_2 \text{Cos}(\emptyset) \text{Cos}(\theta_2) - 2h a_2 \text{Sen}(\emptyset) \text{Sen}(\theta_2) + 2a_2 X_Q \text{Cos}(\theta_2) \\
 & + 2a_2 Y_Q \text{Sen}(\theta_2) - 2h X_Q \text{Cos}(\emptyset) - 2h Y_Q \text{Sen}(\emptyset) \\
 & = b_2^2 (\text{Cos}^2(\theta_2 + \alpha_2) + \text{Sen}^2(\theta_2 + \alpha_2))
 \end{aligned}$$

Reemplazando la identidad: $\text{Cos}^2(\theta_2) + \text{Sen}^2(\theta_2) = 1$

$$\begin{aligned}
 (A.10) \quad & X_A^2 + Y_A^2 - 2X_A X_Q - 2Y_A Y_Q + X_Q^2 + Y_Q^2 + h^2 + a_2^2 - b_2^2 + \\
 & 2X_A h \text{Cos}(\emptyset) + 2Y_A h \text{Sen}(\emptyset) - 2X_A a_2 \text{Cos}(\theta_2) - 2Y_A a_2 \text{Sen}(\theta_2) - \\
 & 2h a_2 \text{Cos}(\emptyset) \text{Cos}(\theta_2) - 2h a_2 \text{Sen}(\emptyset) \text{Sen}(\theta_2) + 2a_2 X_Q \text{Cos}(\theta_2) + 2a_2 Y_Q \text{Sen}(\theta_2) - \\
 & 2h X_Q \text{Cos}(\emptyset) - 2h Y_Q \text{Sen}(\emptyset) = 0
 \end{aligned}$$

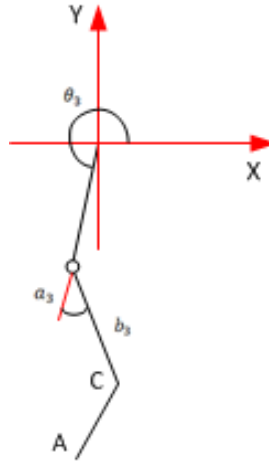


Figura 98. Tercera cadena cinemática

De la tercera Cadena cinemática (figura 101) se tiene que:

$$(A.11) \quad X_C = X_R + a_3 \text{Cos}(\theta_3) + b_3 \text{Cos}(\theta_3 + \alpha_3)$$

$$(A.12) \quad Y_C = Y_R + a_3 \text{Sen}(\theta_3) + b_3 \text{Sen}(\theta_3 + \alpha_3)$$

Reemplazando las ecuaciones A.3 y A.4 en las ecuaciones A.11 y A.12:

$$X_A + h \text{Cos}(\emptyset + \frac{\pi}{3}) = X_R + a_3 \text{Cos}(\theta_3) + b_3 \text{Cos}(\theta_3 + \alpha_3)$$

$$Y_A + h\text{Sen}\left(\phi + \frac{\pi}{3}\right) = Y_R + a_3\text{Sen}(\theta_3) + b_3\text{Sen}(\theta_3 + \alpha_3)$$

Agrupando los términos:

$$X_A + h\text{Cos}\left(\phi + \frac{\pi}{3}\right) - a_3\text{Cos}(\theta_3) - X_R = b_3\text{Cos}(\theta_3 + \alpha_3)$$

$$Y_A + h\text{Sen}\left(\phi + \frac{\pi}{3}\right) - a_3\text{Sen}(\theta_3) - Y_R = b_3\text{Sen}(\theta_3 + \alpha_3)$$

Elevando ambos términos al cuadrado

$$\begin{aligned} X_A^2 + h^2\text{Cos}^2\left(\phi + \frac{\pi}{3}\right) + a_3^2\text{Cos}^2(\theta_3) + X_R^2 + 2X_Ah\text{Cos}\left(\phi + \frac{\pi}{3}\right) - 2X_Aa_3\text{Cos}(\theta_3) \\ - 2X_AX_R - 2ha_3\text{Cos}\left(\phi + \frac{\pi}{3}\right)\text{Cos}(\theta_3) + 2a_3X_R\text{Cos}(\theta_3) \\ - 2hX_R\text{Cos}\left(\phi + \frac{\pi}{3}\right) = b_3^2\text{Cos}^2(\theta_3 + \alpha_3) \end{aligned}$$

$$\begin{aligned} Y_A^2 + h^2\text{Sen}^2\left(\phi + \frac{\pi}{3}\right) + a_3^2\text{Sen}^2(\theta_3) + Y_R^2 + 2Y_Ah\text{Sen}\left(\phi + \frac{\pi}{3}\right) - 2Y_Aa_3\text{Sen}(\theta_3) \\ - 2Y_AY_R - 2ha_3\text{Sen}\left(\phi + \frac{\pi}{3}\right)\text{Sen}(\theta_3) + 2a_3Y_R\text{Sen}(\theta_3) - 2hY_R\text{Sen}\left(\phi + \frac{\pi}{3}\right) \\ = b_3^2\text{Sen}^2(\theta_3 + \alpha_3) \end{aligned}$$

Sumando las dos ecuaciones:

$$\begin{aligned} X_A^2 + Y_A^2 - 2X_AX_R - 2Y_AY_R + X_R^2 + Y_R^2 + h^2(\text{Cos}^2\left(\phi + \frac{\pi}{3}\right) + \text{Sen}^2\left(\phi + \frac{\pi}{3}\right)) \\ + a_3^2(\text{Cos}^2(\theta_3) + \text{Sen}^2(\theta_3)) + 2X_Ah\text{Cos}\left(\phi + \frac{\pi}{3}\right) + 2Y_Ah\text{Sen}\left(\phi + \frac{\pi}{3}\right) \\ - 2X_Aa_3\text{Cos}(\theta_3) - 2Y_Aa_3\text{Sen}(\theta_3) - 2ha_3\text{Cos}\left(\phi + \frac{\pi}{3}\right)\text{Cos}(\theta_3) \\ - 2ha_3\text{Sen}\left(\phi + \frac{\pi}{3}\right)\text{Sen}(\theta_3) + 2a_3X_R\text{Cos}(\theta_3) + 2a_3Y_R\text{Sen}(\theta_3) \\ - 2hX_R\text{Cos}\left(\phi + \frac{\pi}{3}\right) - 2hY_R\text{Sen}\left(\phi + \frac{\pi}{3}\right) \\ = b_3^2(\text{Cos}^2(\theta_3 + \alpha_3) + \text{Sen}^2(\theta_3 + \alpha_3)) \end{aligned}$$

Reemplazando la ecuación anterior con la identidad: $\text{Cos}^2(\theta_2) + \text{Sen}^2(\theta_2) = 1$

$$\begin{aligned}
(A.13) \quad & X_A^2 + Y_A^2 - 2X_A X_R - 2Y_A Y_R + X_R^2 + Y_R^2 + h^2 + a_3^2 - b_3^2 \\
& + 2X_A h \cos\left(\phi + \frac{\pi}{3}\right) + 2Y_A h \sin\left(\phi + \frac{\pi}{3}\right) - 2X_A a_3 \cos(\theta_3) \\
& - 2Y_A a_3 \sin(\theta_3) - 2h a_3 \cos\left(\phi + \frac{\pi}{3}\right) \cos(\theta_3) \\
& - 2h a_3 \sin\left(\phi + \frac{\pi}{3}\right) \sin(\theta_3) + 2a_3 X_R \cos(\theta_3) + 2a_3 Y_R \sin(\theta_3) \\
& - 2h X_R \cos\left(\phi + \frac{\pi}{3}\right) - 2h Y_R \sin\left(\phi + \frac{\pi}{3}\right) = 0
\end{aligned}$$

ANEXO B

Cinemática inversa:

Para Calcular el ángulo θ_1 , se toma la ecuación A.7 de cinemática directa:

$$(A.7) \quad X_A^2 + Y_A^2 - 2X_A a_1 \cos(\theta_1) - 2Y_A a_1 \sin(\theta_1) + a_1^2 - b_1^2 = 0$$

Para facilidad de despeje de los ángulos, se realizaron las siguientes igualdades factorizando la ecuación:

$$e_1 = -2Y_A a_1$$

$$e_2 = -2X_A a_1$$

$$e_3 = X_A^2 + Y_A^2 + a_1^2 - b_1^2$$

Reemplazando usando las igualdades anteriormente enunciadas:

$$e_1 \sin(\theta_1) + e_2 \cos(\theta_1) + e_3 = 0$$

Usando las siguientes identidades trigonométricas:

$$\sin(\theta_1) = \frac{2t_1}{1+t_1^2}$$

$$\cos(\theta_1) = \frac{1-t_1^2}{1+t_1^2}$$

$$t_1 = \tan\left(\frac{\theta_1}{2}\right)$$

Reemplazando las identidades trigonométricas:

$$e_1 \frac{2t_1}{1+t_1^2} + e_2 \frac{1-t_1^2}{1+t_1^2} + e_3 = 0$$

Multiplicando por $1+t_1^2$

$$e_1 \frac{(2t_1)(1+t_1^2)}{1+t_1^2} + e_2 \frac{(1-t_1^2)(1+t_1^2)}{1+t_1^2} + e_3(1+t_1^2) = 0$$

$$e_1(2t_1) + e_2(1-t_1^2) + e_3(1+t_1^2) = 0$$

$$2t_1 e_1 + e_2 - t_1^2 e_2 + e_3 + t_1^2 e_3 = 0$$

$$-t_1^2(-e_2 + e_3) + 2t_1e_1 + e_3 + e_2 = 0$$

Aplicando la fórmula de una ecuación cuadrática: $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

$$t_1 = \frac{-2e_1 \pm \sqrt{4e_1^2 - 4(-e_2 + e_3)(e_2 + e_3)}}{2(-e_2 + e_3)}$$

$$t_1 = \frac{-2e_1 \pm \sqrt{4(e_1^2 - (e_3^2 - e_2^2))}}{2(-e_2 + e_3)}$$

$$t_1 = \frac{-2e_1 \pm \sqrt{4\sqrt{(e_1^2 - e_3^2 + e_2^2)}}}{2(-e_2 + e_3)}$$

$$\theta_i = 2Tan^{-1} \left[\frac{-e_1 \pm \sqrt{(e_1^2 - e_3^2 + e_2^2)}}{-e_2 + e_3} \right]$$

Para Calcular el ángulo θ_2 , se toma la ecuación A.10 de cinemática directa:

$$(A.10) \quad X_A^2 + Y_A^2 - 2X_A X_Q - 2Y_A Y_Q + X_Q^2 + Y_Q^2 + h^2 + a_2^2 - b_2^2 + 2X_A h \cos(\phi) + 2Y_A h \sin(\phi) - 2X_A a_2 \cos(\theta_2) - 2Y_A a_2 \sin(\theta_2) - 2h a_2 \cos(\phi) \cos(\theta_2) - 2h a_2 \sin(\phi) \sin(\theta_2) + 2a_2 X_Q \cos(\theta_2) + 2a_2 Y_Q \sin(\theta_2) - 2h X_Q \cos(\phi) - 2h Y_Q \sin(\phi) = 0$$

Factorizando:

$$\begin{aligned} & \sin(\theta_2) \left(-2Y_A a_2 + 2a_2 Y_Q - 2h a_2 \sin(\phi) \right) \\ & + \cos(\theta_2) \left(-2X_A a_2 + 2a_2 X_Q - 2h a_2 \cos(\phi) \right) + X_A^2 + Y_A^2 + 2X_A X_Q \\ & + X_Q^2 + Y_Q^2 + h^2 + a_2^2 - b_2^2 + 2X_A h \cos(\phi) + 2Y_A h \sin(\phi) \\ & - 2h X_Q \cos(\phi) \end{aligned}$$

Para facilidad de despeje de los ángulos, se realizaron las siguientes igualdades factorizando la ecuación:

$$e_4 = -2Y_A a_2 + 2a_2 Y_Q - 2h a_2 \sin(\phi)$$

$$e_5 = -2X_A a_2 + 2a_2 X_Q - 2h a_2 \cos(\phi)$$

$$e_6 = X_A^2 + Y_A^2 + 2X_A X_Q + X_Q^2 + Y_Q^2 + h^2 + a_2^2 - b_2^2 + 2X_A h \cos(\phi) + 2Y_A h \sin(\phi) - 2h X_Q \cos(\phi)$$

Reemplazando usando las igualdades anteriormente enunciadas:

$$e_4 \text{Sen}(\theta_2) + e_5 \text{Cos}(\theta_2) + e_6 = 0$$

Usando las siguientes identidades trigonométricas:

$$\text{Sen}(\theta_2) = \frac{2t_2}{1+t_2^2}$$

$$\text{Cos}(\theta_2) = \frac{1-t_2^2}{1+t_2^2}$$

$$t_2 = \text{Tan}\left(\frac{\theta_2}{2}\right)$$

Reemplazando las identidades trigonométricas:

$$e_4 \frac{2t_2}{1+t_2^2} + e_5 \frac{1-t_2^2}{1+t_2^2} + e_6 = 0$$

Multiplicando por $1+t_2^2$

$$e_4 \frac{(2t_2)(1+t_2^2)}{1+t_2^2} + e_5 \frac{(1-t_2^2)(1+t_2^2)}{1+t_2^2} + e_6(1+t_2^2) = 0$$

$$e_4(2t_2) + e_5(1-t_2^2) + e_6(1+t_2^2) = 0$$

$$2t_2e_4 + e_5 - t_2^2e_5 + e_6 + t_2^2e_6 = 0$$

$$-t_2^2(-e_5 + e_6) + 2t_2e_4 + e_6 + e_5 = 0$$

Aplicando la fórmula de una ecuación cuadrática: $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

$$t_2 = \frac{-2e_4 \pm \sqrt{4e_4^2 - 4(-e_5 + e_6)(e_5 + e_6)}}{2(-e_5 + e_6)}$$

$$t_2 = \frac{-2e_4 \pm \sqrt{4(e_4^2 - (e_6^2 - e_5^2))}}{2(-e_5 + e_6)}$$

$$t_2 = \frac{-2e_4 \pm \sqrt{4\sqrt{(e_4^2 - e_6^2 + e_5^2)}}}{2(-e_5 + e_6)}$$

$$\theta_2 = 2\text{Tan}^{-1} \left[\frac{-e_4 \pm \sqrt{(e_4^2 - e_6^2 + e_5^2)}}{-e_5 + e_6} \right]$$

Para Calcular el ángulo θ_3 , se toma la ecuación A.13 de cinemática directa:

$$\begin{aligned}
(A.13) \quad & X_A^2 + Y_A^2 - 2X_A X_R - 2Y_A Y_R + X_R^2 + Y_R^2 + h^2 + a_3^2 - b_3^2 \\
& + 2X_A h \cos\left(\phi + \frac{\pi}{3}\right) + 2Y_A h \sin\left(\phi + \frac{\pi}{3}\right) - 2X_A a_3 \cos(\theta_3) \\
& - 2Y_A a_3 \sin(\theta_3) - 2h a_3 \cos\left(\phi + \frac{\pi}{3}\right) \cos(\theta_3) \\
& - 2h a_3 \sin\left(\phi + \frac{\pi}{3}\right) \sin(\theta_3) + 2a_3 X_R \cos(\theta_3) + 2a_3 Y_R \sin(\theta_3) \\
& - 2h X_R \cos\left(\phi + \frac{\pi}{3}\right) - 2h Y_R \sin\left(\phi + \frac{\pi}{3}\right) = 0
\end{aligned}$$

Factorizando y agrupando dichos factores en variables sencillas para realizar el despeje, tenemos que:

$$\begin{aligned}
e_7 &= -2Y_A a_3 + 2a_3 Y_R - 2h a_3 \sin\left(\phi + \frac{\pi}{3}\right) \\
e_8 &= -2X_A a_3 + 2a_3 X_R - 2h a_3 \cos\left(\phi + \frac{\pi}{3}\right) \\
e_9 &= X_A^2 + Y_A^2 + 2X_A X_R + X_R^2 + Y_R^2 + h^2 + a_3^2 - b_3^2 + 2X_A h \cos\left(\phi + \frac{\pi}{3}\right) \\
&+ 2Y_A h \sin\left(\phi + \frac{\pi}{3}\right) - 2h X_R \cos\left(\phi + \frac{\pi}{3}\right) - 2h Y_R \sin\left(\phi + \frac{\pi}{3}\right)
\end{aligned}$$

Reemplazando usando las igualdades anteriormente enunciadas:

$$e_7 \sin(\theta_3) + e_8 \cos(\theta_3) + e_9 = 0$$

Usando las siguientes identidades trigonométricas:

$$\begin{aligned}
\sin(\theta_3) &= \frac{2t_3}{1 + t_3^2} \\
\cos(\theta_3) &= \frac{1 - t_3^2}{1 + t_3^2} \\
t_3 &= \tan\left(\frac{\theta_3}{2}\right)
\end{aligned}$$

Reemplazando las identidades trigonométricas:

$$e_7 \frac{2t_3}{1 + t_3^2} + e_8 \frac{1 - t_3^2}{1 + t_3^2} + e_9 = 0$$

Multiplicando por $1 + t_3^2$

$$e_7 \frac{(2t_3)(1+t_3^2)}{1+t_3^2} + e_8 \frac{(1-t_3^2)(1+t_3^2)}{1+t_3^2} + e_9(1+t_3^2) = 0$$

$$e_7(2t_3) + e_8(1-t_3^2) + e_9(1+t_3^2) = 0$$

$$2t_3e_7 + e_8 - t_3^2e_8 + e_9 + t_3^2e_9 = 0$$

$$-t_3^2(-e_8 + e_9) + 2t_3e_7 + e_9 + e_8 = 0$$

Aplicando la fórmula de una ecuación cuadrática: $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

$$t_3 = \frac{-2e_7 \pm \sqrt{4e_7^2 - 4(-e_8 + e_9)(e_8 + e_9)}}{2(-e_8 + e_9)}$$

$$t_3 = \frac{-2e_7 \pm \sqrt{4(e_7^2 - (e_9^2 - e_8^2))}}{2(-e_8 + e_9)}$$

$$t_3 = \frac{-2e_7 \pm \sqrt{4\sqrt{(e_7^2 - e_9^2 + e_8^2)}}}{2(-e_8 + e_9)}$$

$$\theta_3 = 2Tan^{-1} \left[\frac{-e_7 \pm \sqrt{(e_7^2 - e_9^2 + e_8^2)}}{-e_8 + e_9} \right]$$

ANEXO C

Cinemática directa:

De acuerdo a las ecuaciones halladas por medio del análisis de la geometría del mecanismo, podemos encontrar una ecuación general que tenga como única variable ϕ en función de variables conocidas como las longitudes de los eslabones $a_1 \dots a_6$, las coordenadas del efector final X_A, Y_A y los ángulos $\theta_1, \theta_2, \theta_3$:

$$(C.1) \quad X_A^2 + Y_A^2 + e_{11}X_A + e_{12}Y_A + e_{13} = 0$$

$$(C.2) \quad X_A^2 + Y_A^2 + e_{21}X_A + e_{22}Y_A + e_{23} = 0$$

$$(C.3) \quad X_A^2 + Y_A^2 + e_{31}X_A + e_{32}Y_A + e_{33} = 0$$

Donde:

$$e_{11} = -2a_1 \cos(\theta_1)$$

$$e_{12} = -2a_1 \sin(\theta_1)$$

$$e_{13} = a_1^2 - b_1^2$$

$$e_{21} = -2X_Q + 2h \cos(\phi) - 2a_2 \cos(\theta_2)$$

$$e_{22} = -2Y_Q + 2h \sin(\phi) - 2a_2 \sin(\theta_2)$$

$$e_{23} = X_Q^2 + Y_Q^2 + h^2 + a_2^2 - b_2^2 - 2a_2 h \cos(\theta_2) \cos(\phi) - 2a_2 h \sin(\theta_2) \sin(\phi) - 2X_Q h \cos(\phi) - 2Y_Q h \sin(\phi) + 2X_Q a_2 \cos(\theta_2) + 2Y_Q a_2 \sin(\theta_2)$$

$$e_{31} = -2X_R + 2h \cos\left(\phi + \frac{\pi}{3}\right) - 2a_3 \cos(\theta_3)$$

$$e_{32} = -2Y_R + 2h \sin\left(\phi + \frac{\pi}{3}\right) - 2a_3 \sin(\theta_3)$$

$$e_{33} = X_R^2 + Y_R^2 + h^2 + a_3^2 - b_3^2 - 2a_3 h \cos(\theta_3) \cos\left(\phi + \frac{\pi}{3}\right) - 2a_3 h \sin(\theta_3) \sin\left(\phi + \frac{\pi}{3}\right) - 2X_R h \cos\left(\phi + \frac{\pi}{3}\right) - 2Y_R h \sin\left(\phi + \frac{\pi}{3}\right) + 2X_R a_3 \cos(\theta_3) + 2Y_R a_3 \sin(\theta_3)$$

En el Sistema de ecuaciones enunciado anteriormente, se puede ver que encontramos, 4 variables desconocidas $[\theta_1, \theta_2, \theta_3, \phi]$, de las cuales gracias a la cinemática inversa, se pueden calcular los ángulos θ_1, θ_2 y θ_3 que describen las articulaciones principales con base al plano (X, Y) , que se designaron para el robot de tal manera que solo tenemos una incógnita ϕ , pero debemos hallar esta variable en función de los tres ángulo principales

Restando la ecuación C.2 de la C.1, tenemos que:

$$(C.4) \quad e'_{11}X_A + e'_{12}Y_A + e'_{13} = 0$$

$$e'_{11} = e_{11} - e_{21}$$

$$e'_{12} = e_{12} - e_{22}$$

$$e'_{13} = e_{13} - e_{23}$$

Restando la ecuación C.3 de la C.1, tenemos que:

$$(C.5) \quad e'_{21}X_A + e'_{22}Y_A + e'_{23} = 0$$

$$e'_{21} = e_{11} - e_{31}$$

$$e'_{22} = e_{12} - e_{32}$$

$$e'_{23} = e_{13} - e_{33}$$

Tomando la ecuación C.1, C.4 y C.5, se resuelve en la ecuación C.4 y C.5 en términos de X_A y Y_A respectivamente y se despejan en la ecuación C.1, lo que nos da una ecuación en términos de δ de cuarto grado, de la siguiente forma:

$$\delta_1^2 + \delta_2^2 + e_{11}\partial\delta_1 + e_{12}\partial\delta_2 + e_{13}\partial^2 = 0$$

$$\partial = e'_{11}e'_{22} - e'_{12}e'_{21}$$

$$\delta_1 = e'_{12}e'_{23} - e'_{13}e'_{22}$$

$$\delta_2 = e'_{13}e'_{21} - e'_{11}e'_{23}$$

De tal forma que si aproximamos la ecuación de la cinemática directa, usando los pasos enunciados en el presente anexo:

$$\begin{aligned} \delta_1^2 + \delta_2^2 + e_{11}\partial\delta_1 + e_{12}\partial\delta_2 + e_{13}\partial^2 &= 0 \\ \rightarrow (e'_{12}e'_{23} - e'_{13}e'_{22})^2 + (e'_{13}e'_{21} - e'_{11}e'_{23})^2 \\ + e_{11}(e'_{11}e'_{22} - e'_{12}e'_{21})(e'_{12}e'_{23} - e'_{13}e'_{22}) + e_{12}(e'_{11}e'_{22} \\ - e'_{12}e'_{21})(e'_{13}e'_{21} - e'_{11}e'_{23}) + e_{13}(e'_{11}e'_{22} - e'_{12}e'_{21})^2 &= 0 \end{aligned}$$

Solucionando los productos notables:

$$\begin{aligned}
&\rightarrow ((e'_{12}e'_{23})^2 - (2e'_{12}e'_{23}e'_{13}e'_{22}) + (e'_{13}e'_{22})^2) \\
&\quad + ((e'_{13}e'_{21})^2 - (2e'_{13}e'_{21} - e'_{11}e'_{23}) + (e'_{11}e'_{23})^2) \\
&\quad + e_{11}[(e'_{11}e'_{22}e'_{12}e'_{23}) - (e'_{11}e'_{22}e'_{13}e'_{22}) - (e'_{12}e'_{21}e'_{13}e'_{21}) \\
&\quad + (e'_{12}e'_{21}e'_{11}e'_{23})] \\
&\quad + e_{12}[(e'_{11}e'_{22}e'_{13}e'_{21}) - (e'_{11}e'_{22}e'_{11}e'_{23}) - (e'_{12}e'_{21}e'_{13}e'_{21}) \\
&\quad + (e'_{12}e'_{21}e'_{11}e'_{23})] + e_{13}[(e'_{13}e'_{21})^2 - (2e'_{13}e'_{21}e'_{11}e'_{23}) + (e'_{11}e'_{23})^2] \\
&= 0
\end{aligned}$$

Simplificando:

$$\begin{aligned}
&(e'_{12}e'_{23})^2 - (2e'_{12}e'_{23}e'_{13}e'_{22}) + (e'_{13}e'_{22})^2 \\
&\quad + (e'_{13}e'_{21})^2 - (2e'_{13}e'_{21} - e'_{11}e'_{23}) + (e'_{11}e'_{23})^2 + e_{11}e'_{11}e'_{22}e'_{12}e'_{23} \\
&\quad - e_{11}e'_{11}e'_{22}e'_{13}e'_{22} - e_{11}e'_{12}e'_{21}e'_{13}e'_{21} + e_{11}e'_{12}e'_{21}e'_{11}e'_{23} \\
&\quad + e_{12}e'_{11}e'_{22}e'_{13}e'_{21} - e_{12}e'_{11}e'_{22}e'_{13} - e_{12}e'_{21}e'_{12}e'_{13} + e_{12}e'_{12}e'_{21}e'_{11}e'_{23} \\
&\quad + e_{13}e'_{13}e'_{21}e'_{11}e'_{23} - 2e_{13}e'_{13}e'_{21}e'_{11}e'_{23} + e_{13}e'_{11}e'_{23}e'_{11}e'_{23} = 0
\end{aligned}$$

Reemplazando:

$$\begin{aligned}
(C.6) \quad &((e_{12} - e_{22})^2(e_{13} - e_{33})^2) \\
&- 2[(e_{12} - e_{22})(e_{13} - e_{33})(e_{13} - e_{23})(e_{12} - e_{32})] \\
&+ ((e_{13} - e_{22})^2(e_{12} - e_{33})^2) + ((e_{13} - e_{23})^2(e_{11} - e_{31})^2) \\
&- 2[(e_{13} - e_{23})(e_{11} - e_{31})(e_{11} - e_{21})(e_{13} - e_{33})] \\
&+ ((e_{11} - e_{21})^2(e_{13} - e_{33})^2) \\
&+ e_{11}[(e_{11} - e_{21})(e_{12} - e_{32})(e_{12} - e_{22})(e_{13} - e_{33})] \\
&- e_{11}[(e_{11} - e_{21})(e_{12} - e_{32})(e_{13} - e_{23})(e_{12} - e_{33})] \\
&- e_{11}[(e_{12} - e_{22})(e_{11} - e_{31})(e_{12} - e_{22})(e_{13} - e_{33})] \\
&+ e_{11}[(e_{12} - e_{22})(e_{11} - e_{31})(e_{13} - e_{23})(e_{12} - e_{32})] \\
&+ e_{12}[(e_{11} - e_{21})(e_{12} - e_{32})(e_{13} - e_{23})(e_{11} - e_{31})] \\
&+ e_{12}[(e_{11} - e_{21})^2(e_{12} - e_{32})(e_{13} - e_{33})] \\
&- e_{12}[(e_{11} - e_{31})^2(e_{12} - e_{22})(e_{13} - e_{23})] \\
&+ e_{12}[(e_{12} - e_{22})(e_{11} - e_{31})(e_{11} - e_{21})(e_{13} - e_{33})] \\
&+ e_{13}[(e_{13} - e_{23})^2(e_{11} - e_{31})^2] \\
&- 2e_{13}[(e_{13} - e_{23})(e_{11} - e_{31})(e_{11} - e_{21})(e_{13} - e_{33})] \\
&+ e_{13}[(e_{11} - e_{21})^2(e_{13} - e_{33})^2] = 0
\end{aligned}$$

Como se ha venido desarrollando con anterioridad, se despejan las ecuaciones $e_{11} \dots e_{33}$ usando la siguiente identidad

$$\text{Sen}(\theta_i) = \frac{2t_i}{1 + t_i^2}$$

$$\cos(\theta_i) = \frac{1 - t_i^2}{1 + t_i^2}$$

$$t_i = \tan\left(\frac{\theta_i}{2}\right)$$

Con $i = 1, 2$ y 3

Sustituyendo en $e_{11} \dots e_{33}$:

$$e_{11} = -2a_1 \left[\frac{1 - \tan\left(\frac{\theta_1}{2}\right)^2}{1 + \tan\left(\frac{\theta_1}{2}\right)^2} \right]$$

$$e_{12} = -2a_1 \left[\frac{2 \tan\left(\frac{\theta_2}{2}\right)}{1 + \tan\left(\frac{\theta_2}{2}\right)^2} \right]$$

$$e_{13} = a_1^2 - b_1^2$$

$$e_{21} = -2X_Q + 2h \cos(\phi) - 2a_2 \left[\frac{1 - \tan\left(\frac{\theta_2}{2}\right)^2}{1 + \tan\left(\frac{\theta_2}{2}\right)^2} \right]$$

$$e_{22} = -2Y_Q + 2h \sin(\phi) - 2a_2 \left[\frac{2 \tan\left(\frac{\theta_2}{2}\right)}{1 + \tan\left(\frac{\theta_2}{2}\right)^2} \right]$$

$$\begin{aligned} e_{23} = & X_Q^2 + Y_Q^2 + h^2 + a_2^2 - b_2^2 - 2a_2 h \left[\frac{1 - \tan\left(\frac{\theta_2}{2}\right)^2}{1 + \tan\left(\frac{\theta_2}{2}\right)^2} \right] \cos(\phi) \\ & - 2a_2 h \left[\frac{2 \tan\left(\frac{\theta_2}{2}\right)}{1 + \tan\left(\frac{\theta_2}{2}\right)^2} \right] \sin(\phi) - 2X_Q h \cos(\phi) - 2Y_Q h \sin(\phi) \\ & + 2X_Q a_2 \left[\frac{1 - \tan\left(\frac{\theta_2}{2}\right)^2}{1 + \tan\left(\frac{\theta_2}{2}\right)^2} \right] + 2Y_Q a_2 \left[\frac{2 \tan\left(\frac{\theta_2}{2}\right)}{1 + \tan\left(\frac{\theta_2}{2}\right)^2} \right] \end{aligned}$$

$$e_{31} = -2X_R + 2h \cos\left(\phi + \frac{\pi}{3}\right) - 2a_3 \left[\frac{1 - \tan\left(\frac{\theta_3}{2}\right)^2}{1 + \tan\left(\frac{\theta_3}{2}\right)^2} \right]$$

$$e_{32} = -2Y_R + 2h \sin\left(\phi + \frac{\pi}{3}\right) - 2a_3 \left[\frac{2 \tan\left(\frac{\theta_3}{2}\right)}{1 + \tan\left(\frac{\theta_3}{2}\right)^2} \right]$$

$$\begin{aligned}
e_{33} = & X_R^2 + Y_R^2 + h^2 + a_3^2 - b_3^2 - 2a_3h \left[\frac{1 - \tan\left(\frac{\theta_3}{2}\right)^2}{1 + \tan\left(\frac{\theta_3}{2}\right)^2} \right] \cos\left(\phi + \frac{\pi}{3}\right) \\
& - 2a_3h \left[\frac{2\tan\left(\frac{\theta_3}{2}\right)}{1 + \tan\left(\frac{\theta_3}{2}\right)^2} \right] \sin\left(\phi + \frac{\pi}{3}\right) - 2X_Rh \cos\left(\phi + \frac{\pi}{3}\right) \\
& - 2Y_Rh \sin\left(\phi + \frac{\pi}{3}\right) + 2X_Ra_3 \left[\frac{1 - \tan\left(\frac{\theta_3}{2}\right)^2}{1 + \tan\left(\frac{\theta_3}{2}\right)^2} \right] + 2Y_R \left[\frac{2\tan\left(\frac{\theta_3}{2}\right)}{1 + \tan\left(\frac{\theta_3}{2}\right)^2} \right]
\end{aligned}$$

Reemplazando las ecuaciones $e_{11} \dots e_{33}$ en la ecuación 6, tenemos que, el polinomio resultante es uno de octavo grado en función de una incógnita ϕ , que concuerda con las 8 posibles raíces o soluciones que puede generar el sistema para llegar a la ubicación deseada, y que es de la forma:

$$\begin{aligned}
(C.7) \quad & C_0T^0 + C_1T^1 + C_2T^2 + \dots + C_nT^n = 0 \\
& n = 8
\end{aligned}$$

Cada una de las constantes esta en términos de los parámetros geométricos del prototipo, que son las longitudes de los eslabones $a_1 \dots a_6$, las coordenadas del efector final X_A, Y_A y los ángulos $\theta_1, \theta_2, \theta_3$:

ANEXO D

Análisis del jacobiano:

Según se diseñó el sistema (Véase la figura 60), los puntos P, Q y R forman un triángulo equilátero, de lo que se puede concluir:

$$\overline{PQ} = \overline{QR} = \overline{RP} = c$$

Teniendo en cuenta que el triángulo base es un triángulo equilátero, se puede deducir que:

$$\overline{AB} = \overline{BC} = \overline{CA} = h$$

Para el primer brazo del robot (Figura 102):

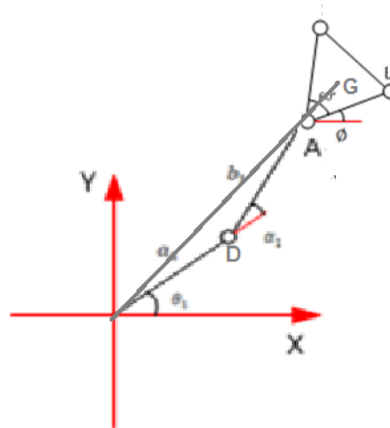


Figura 99. Jacobiano

$$\overline{PG} + \overline{GA} = \overline{PD} + \overline{DA}$$

$$X_G + \phi e_1 = \theta_1 a_1 + (\theta_1 a_1) b_1$$

Derivando la ecuación anterior respecto al tiempo:

$$V_G + \dot{\phi}(kx e_1) = \dot{\theta}_1(kx a_1) + (\dot{\theta}_1 + \dot{a}_1)(kx b_1)$$

Para eliminar a_1 debido a que es una junta pasiva sobre la cual no tenemos control, se multiplica toda la ecuación por b_1

$$\begin{aligned}
b_1 \cdot V_G + \dot{\phi} k \cdot (e_1 x b_1) &= \dot{\theta}_1 k \cdot (a_1 x b_1) \\
b_1 \cdot V_G + \dot{\phi} k \cdot (e_1 x b_1) - \dot{\theta}_1 k \cdot (a_1 x b_1) &= 0 \\
b_{1X} \cdot V_{GX} + b_{1Y} \cdot V_{GY} + \dot{\phi} (e_{1X} b_{1Y} - e_{1Y} b_{1X}) &= \dot{\theta}_1 (a_{1X} b_{1Y} - a_{1Y} b_{1X})
\end{aligned}$$

Representándolo en forma matricial:

$$\begin{bmatrix} b_{1X} & b_{1Y} & e_{1X} b_{1Y} - e_{1Y} b_{1X} \end{bmatrix} \begin{bmatrix} V_{GX} \\ V_{GY} \\ \dot{\phi} \end{bmatrix} = [a_{1X} b_{1Y} - a_{1Y} b_{1X}] [\dot{\theta}_1]$$

Se aplica el procedimiento anterior para las otras cadenas y llegamos a la siguiente expresión en forma matricial:

$$\begin{bmatrix} b_{1X} & b_{1Y} & e_{1X} b_{1Y} - e_{1Y} b_{1X} \\ b_{2X} & b_{2Y} & e_{2X} b_{2Y} - e_{2Y} b_{2X} \\ b_{3X} & b_{3Y} & e_{3X} b_{3Y} - e_{3Y} b_{3X} \end{bmatrix} \begin{bmatrix} V_{GX} \\ V_{GY} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} a_{1X} b_{1Y} - a_{1Y} b_{1X} & 0 & 0 \\ 0 & a_{2X} b_{2Y} - a_{2Y} b_{2X} & 0 \\ 0 & 0 & a_{3X} b_{3Y} - a_{3Y} b_{3X} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}$$

De lo que se puede concluir:

$$\begin{aligned}
J_X &= \begin{bmatrix} b_{1X} & b_{1Y} & e_{1X} b_{1Y} - e_{1Y} b_{1X} \\ b_{2X} & b_{2Y} & e_{2X} b_{2Y} - e_{2Y} b_{2X} \\ b_{3X} & b_{3Y} & e_{3X} b_{3Y} - e_{3Y} b_{3X} \end{bmatrix} \\
J_Q &= \begin{bmatrix} a_{1X} b_{1Y} - a_{1Y} b_{1X} & 0 & 0 \\ 0 & a_{2X} b_{2Y} - a_{2Y} b_{2X} & 0 \\ 0 & 0 & a_{3X} b_{3Y} - a_{3Y} b_{3X} \end{bmatrix}
\end{aligned}$$

$$J_X \dot{x} = J_Q \dot{q}$$

Como:

$$J = J_Q^{-1} J_X \rightarrow \dot{q} = J \dot{x}$$

De tal forma que así podremos conocer las velocidades de cada articulación a partir de una velocidad del eslabón acoplador

Finalmente, se puede conocer la velocidad del eslabón acoplador a partir de las velocidades de las articulaciones con la siguiente ecuación:

$$J \dot{q} = \dot{x}$$

ANEXO E

Código:

BOTON CINEMATICA INVERSA

```
function CI_Callback(hObject, eventdata, handles)

%Selección del eje
axes(handles.axes5);
cla;
set(handles.text21, 'String', '');
%Visualización modo ON cuadros de texto de ángulo theta 1,
theta 2 y theta 3
set(handles.edit10, 'Visible', 'on')
set(handles.edit11, 'Visible', 'on')
set(handles.edit12, 'Visible', 'on')
%Obtención de variables dadas por el usuario de forma
automática o manual
L1=str2double(get(handles.A, 'string'));
L2=str2double(get(handles.B, 'string'));
Efx=str2double(get(handles.XG, 'string'));
Efy=str2double(get(handles.YG, 'string'));
phi=str2double(get(handles.FI, 'string'));
%Obtención de dato de la configuración del robot
M=str2double(get(handles.text22, 'string'));
%Parámetros para el envío de datos hacia los servos
install_arduino
a=arduino('COM4');
%Configuración de pines PWM del arduino para enviar los
ángulos a los servos
a.servoAttach(9);
a.servoAttach(10);
a.servoAttach(11);
%Longitud de los lados del triángulo equilátero
h=5;
%Coordenadas del Efector final primer brazo en función de
las coordenadas
%dadas por el usuario
XA=Efx-2.88*cosd(30+phi)
YA=Efy-2.88*sind(30+phi)
XB=XA+(h*cosd(phi));
YB=YA+(h*sind(phi));
XC=XA+(h*cosd(phi+(60)));
YC=YA+(h*sind(phi+(60)));
%Condicionales de configuración del robot
if M==1
e1=-(2*YA*L1)
e2=-(2*XA*L1)
e3=(XA^2)+(YA^2)+(L1^2)-(L2^2)
```

```

%Condicional para mensaje de error en la ubicación del
robot
if (e3^2)>(e2^2)+(e1^2)
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    hold on;
    return
end
%Obtencion ángulo theta 1
Ang1=2*(atand((-e1)+sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2))
%Función piso para el ángulo theta 1, debido a que el solo
se pueden
%comunicar valores enteros al servomotor
if Ang1<0
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2))
Ang1=floor(Ang1);
end
if Ang1>180
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2))
Ang1=floor(Ang1);
end

%Impresion de theta en la interfaz
set(handles.edit10,'string',Ang1);
%Ecuación de la primera coordenada del brazo 1, para
visualizar en la
%simulación eslabón de color azul
XA1=L1*cosd(Ang1)
YA1=L1*sind(Ang1)
h=5;
XQ=30;
e4=- (2*YA*L1) - (2*L1*h*sind(phi));
e5=- (2*XA*L1) - (2*L1*h*cosd(phi)) + (2*XQ*L1);
e6=(XA^2)+(YA^2) - (2*XA*XQ) + (XQ^2) + (h^2) + (L1^2) -
(L2^2) + (2*XA*h*cosd(phi)) + (2*YA*h*sind(phi)) -
(2*XQ*h*cosd(phi));
if (e6^2)>(e4^2)+(e5^2)
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    hold on;
    return
end
%Obtencion angulo theta 2
Ang2=2*(atand((-e4)+sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5))
Ang2=floor(Ang2);
if Ang2<0
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5))
Ang2=floor(Ang2);
end

```

```

if Ang2>180
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5))
Ang2=floor(Ang2);
end
set(handles.edit11,'string',Ang2);
%Ecuación de la primera coordenada del brazo 2, para
visualizar en la
%simulación eslabón de color azul
XA2=XQ+L1*cosd(Ang2)
YA2=L1*sind(Ang2)
YR=25.98;
XR=15;
e7=(-2*YA*L1)+(2*YR*L1)-(2*L1*h*sind(phi+60));
e8=(-2*XA*L1)+(2*XR*L1)-(2*L1*h*cosd(phi+60));
e9=(XA^2)+(YA^2)-(2*XA*XR)-
(2*YA*YR)+(XR^2)+(YR^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi+60))+(2*YA*h*sind(phi+60))-
(2*XR*h*cosd(phi+60))-(2*YR*h*sind(phi+60));
if (e9^2)>(e7^2)+(e8^2)
set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
hold on;
return
end
%Obtencion del angulo 3
Ang3=2*(atand((-e7)+sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8))
Ang31=180+(2*(atand((-e7)+sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);
if Ang3<0
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8))
Ang31=180+(2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);
end
if Ang3>180
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8))
Ang31=180+(2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);
end
%Angulo para enviar al servo número 3, debido a que se
necesita uno que
%gire 360°

set(handles.edit12,'string',Ang3);
%Envio de angulos al arduino a los pines configurados
a.servoWrite(9,Ang1);
a.servoWrite(10,Ang2);

```

```

a.servoWrite(11,Ang31);
%Ecuación de la primera coordenada del brazo 3 , para
visualizar en la
%simulación eslabón de color azul
XA3=XR+L1*cosd(Ang3)
YA3=YR+L1*sind(Ang3)
hold on;
%Graficacion de todos los eslabones y brazos del prototipo
plot(handles.axes5,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes5,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes5,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes5,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes5,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes5,[XA3 XC],[YA3 YC],'r-','linewidth',2);
set(handles.axes5,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
title('ROBOT PARALELO PLANAR RRR')
xlabel('X')
ylabel('Y')
plot(handles.axes5,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes5,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes5,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes5,Efx,Efy,'ko','linewidth',2);
hold on;
end
%Condicionante para la segunda configuración del prototipo
if M==2
e1=-(2*YA*L1)
e2=-(2*XA*L1)
e3=(XA^2)+(YA^2)+(L1^2)-(L2^2)
if (e3^2)>(e1^2)+(e2^2)
    set(handles.text21,'String','El robot no puede alcanzar
la posicion deseada')
    hold on;
    return
end
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2))
Ang1=floor(Ang1);
set(handles.edit10,'string',Ang1);
XA1=L1*cosd(Ang1)
YA1=L1*sind(Ang1)
h=5;
XQ=30;
e4=-(2*YA*L1)-(2*L1*h*sind(phi));
e5=-(2*XA*L1)-(2*L1*h*cosd(phi))+(2*XQ*L1);
e6=(XA^2)+(YA^2)-(2*XA*XQ)+(XQ^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi))+(2*YA*h*sind(phi))-
(2*XQ*h*cosd(phi));
if (e6^2)>(e4^2)+(e5^2)

```

```

        set(handles.text21, 'String', 'El robot no puede alcanzar
la posicion deseada')
        hold on;
        return
end
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5))
Ang2=floor(Ang2);
set(handles.edit11, 'string', Ang2);
XA2=XQ+L1*cosd(Ang2)
YA2=L1*sind(Ang2)
YR=25.98;
XR=15;
e7=(-2*YA*L1)+(2*YR*L1)-(2*L1*h*sind(phi+60));
e8=(-2*XA*L1)+(2*XR*L1)-(2*L1*h*cosd(phi+60));
e9=(XA^2)+(YA^2)-(2*XA*XR)-
(2*YA*YR)+(XR^2)+(YR^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi+60))+(2*YA*h*sind(phi+60))-
(2*XR*h*cosd(phi+60))-(2*YR*h*sind(phi+60));
if (e9^2)>(e7^2)+(e8^2)
    set(handles.text21, 'String', 'El robot no puede alcanzar
la posicion deseada')
    hold on;
    return
end
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8))
Ang3=floor(Ang3);
Ang31=180+(2*(atand((-e7)+sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);
set(handles.edit12, 'string', Ang3);
a.servoWrite(9, Ang1);
a.servoWrite(10, Ang2);
a.servoWrite(11, Ang31);
XA3=XR+L1*cosd(Ang3)
YA3=YR+L1*sind(Ang3)
hold on;
plot(handles.axes5, [0 XA1], [0 YA1], 'b-', 'linewidth', 2);
plot(handles.axes5, [XA1 XA], [YA1 YA], 'r-', 'linewidth', 2);
plot(handles.axes5, [XQ XA2], [0 YA2], 'b-', 'linewidth', 2);
plot(handles.axes5, [XA2 XB], [YA2 YB], 'r-', 'linewidth', 2);
plot(handles.axes5, [XR XA3], [YR YA3], 'b-', 'linewidth', 2);
plot(handles.axes5, [XA3 XC], [YA3 YC], 'r-', 'linewidth', 2);
set(handles.axes5, 'xgrid', 'on', 'ygrid', 'on', 'xlim', [-
5, 35], 'ylim', [-10, 30])
title('ROBOT PARALELO PLANAR RRR')
xlabel('X')
ylabel('Y')
plot(handles.axes5, [XA XB], [YA YB], 'y-', 'linewidth', 2);
plot(handles.axes5, [XB XC], [YB YC], 'y-', 'linewidth', 2);

```

```

plot(handles.axes5,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes5,Efx,Efy,'ko','linewidth',2);
hold on;
end

```

BOTON AREA DE TRABAJO

```

function AT_Callback(hObject, eventdata, handles)

%Rutina para calcular el area de trabajo
%Selección del eje a usar
axes(handles.axes4);
set(handles.text21,'String','')
%Creación de vectores para almacenamiento
XA1=[];
YA1=[];
%Obtención de las longitudes de los eslabones
L1=11.5%str2double(get(handles.A,'string'));
L2=11.5%str2double(get(handles.B,'string'));
Cont=0;
for theta=9:51%Intervalo y grado maximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes4,XA1(Cont),YA1(Cont),'r.','linewidth',2)
set(handles.axes4,'xgrid','on','ygrid','on','Xlim',[0,35],'
Ylim',[0,30])%Delimitación de las coordenadas del eje
title('AREA DE TRABAJO')
xlabel('X')
ylabel('Y')
hold on
end
%Ubicación del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];
YB1=[];
Cont1=0;
for theta2=129:171%Intervalo y grado maximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]

```



```

plot(handles.axes4, XB1 (Cont1), YB1 (Cont1), 'g.', 'linewidth', 2
)
set(handles.axes4, 'xgrid', 'on', 'ygrid', 'on', 'Xlim', [0, 35], '
Ylim', [0, 30])
hold on
end
%Ubicación del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=249:291 %Intervalo y grado máximo del ángulo
theta 3
Cont2=Cont2+1;
XC=XR+ (L1*cosd(theta3))+(L2*cosd(theta3));
YC=YR+ (L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes4, XC1 (Cont2), YC1 (Cont2), 'b.', 'linewidth', 2
)
set(handles.axes4, 'xgrid', 'on', 'ygrid', 'on', 'Xlim', [-
5, 35], 'Ylim', [-10, 30])
hold on
end
%Obtención de solo una coordenada en el plano donde se
grafica el área de
%trabajo por medio del cursor del mouse
[X1 Y1]=ginput(1)
set(handles.XG, 'string', X1);
set(handles.YG, 'string', Y1);

```

BOTON SELECCIONAR PUNTO

```

function pushbutton7_Callback(hObject, eventdata, handles)
%Rutina del botón seleccionar
%Selección del eje a usar
axes(handles.axes4);
set(handles.text21, 'String', '')
%Comando para obtener la coordenada que se seleccione en el
eje
[X1 Y1]=ginput(1);
%Almacenamiento de las coordenadas en la interfaz grafica
set(handles.XG, 'string', X1);
set(handles.YG, 'string', Y1);

```

BOTON LINEA RECTA PASOS

```

function pushbutton8_Callback(hObject, eventdata, handles)
%Rutina para la línea recta por pasos

```

```

%Selección del eje donde se graficara
axes(handles.axes5);
set(handles.text21,'String','')
%Limpieza del eje antes de graficar
cla;
%Activar visualización de los ángulos
set(handles.edit10,'Visible','on')
set(handles.edit11,'Visible','on')
set(handles.edit12,'Visible','on')
%Obtención de las longitudes de los eslabones, coordenadas
y ángulo phi de
%la interfaz
L1=str2double(get(handles.A,'string'));
L2=str2double(get(handles.B,'string'));
Lim=str2double(get(handles.Linea,'string'));
Efy=str2double(get(handles.YG,'string'));
phi=str2double(get(handles.FI,'string'));
%Obtención del tipo de configuración requerida para el
robot
M=str2double(get(handles.text22,'string'));
h=5;
Cont=0;
YA=Efy-2.88*sind(30+phi)
set(handles.text21,'String','')
install_arduino
a=arduino('COM4');
%Configuración de pines PWM del arduino para enviar los
ángulos a los
%servos
a.servoAttach(9);
a.servoAttach(10);
a.servoAttach(11);
%Creación de vectores para almacenamiento de coordenadas X
y Y
Vx=[];
Vy=[];
%Almacenamiento de las coordenadas iniciales en el vector
VX y VY
%Vx=[Vx Efx];
Vy=[Vy Efy];
for i=1:Lim % Intervalo y punto máximo de X
Cont=Cont+1;
Efx=str2double(get(handles.XG,'string'));
Efx=Efx-i;%Movimiento de derecha a izquierda
XA=Efx-2.88*cosd(30+phi);%Obtención de coordenada en X
inicial
%XA=Efx;
%Almacenamiento de coordenadas
Vx=[Vx Efx]

```

```

Vy=[Vy Efy]
XB=XA+(h*cos(phi));
YB=YA+(h*sind(phi));
XC=XA+(h*cosd(phi+(60)));
YC=YA+(h*sind(phi+(60)));
%Condicionante para la primera configuración del robot
if M==1
e1=-(2*YA*L1);
e2=-(2*XA*L1);
e3=(XA^2)+(YA^2)+(L1^2)-(L2^2);
%Condicionante para mensaje de error, si el robot no puede
llegar graficara
%hasta la última posición que pudo alcanzar
if (e3^2)>(e2^2)+(e1^2)
    hold on;
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    break
end
%Cálculo del ángulo theta 1
Ang1=2*(atand((-e1)+sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
if Ang1<0
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
Ang1=floor(Ang1);
end
if Ang1>180
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
Ang1=floor(Ang1);
end
set(handles.edit10,'string',Ang1);
XA1=L1*cosd(Ang1);
YA1=L1*sind(Ang1);
h=5;
XQ=30;
e4=-(2*YA*L1)-(2*L1*h*sind(phi));
e5=-(2*XA*L1)-(2*L1*h*cosd(phi))+(2*XQ*L1);
e6=(XA^2)+(YA^2)-(2*XA*XQ)+(XQ^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi))+(2*YA*h*sind(phi))-
(2*XQ*h*cosd(phi));
if (e6^2)>(e5^2)+(e4^2)
    hold on;
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    break
    %return
end
Ang2=2*(atand((-e4)+sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
if Ang2<0
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));

```

```

Ang2=floor (Ang2);
end
if Ang2>180
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
Ang2=floor (Ang2);
end
set(handles.edit11,'string',Ang2);
XA2=XQ+L1*cosd(Ang2);
YA2=L1*sind(Ang2);
YR=25.98;
XR=15;
e7=(-2*YA*L1)+(2*YR*L1)-(2*L1*h*sind(phi+60));
e8=(-2*XA*L1)+(2*XR*L1)-(2*L1*h*cosd(phi+60));
e9=(XA^2)+(YA^2)-(2*XA*XR)-
(2*YA*YR)+(XR^2)+(YR^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi+60))+(2*YA*h*sind(phi+60))-
(2*XR*h*cosd(phi+60))-(2*YR*h*sind(phi+60));
if (e9^2)>(e7^2)+(e8^2)
    hold on;
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    %Detiene el programa
    break
end
Ang3=2*(atand((-e7)+sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
Ang31=180+(2*(atand((-e7)+sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor (Ang31);
if Ang3<0
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
Ang31=180+(2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor (Ang31);
end
if Ang3>180
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
Ang31=180+(2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor (Ang31);
end
set(handles.edit12,'string',Ang3);
a.servoWrite(9,Ang1);
a.servoWrite(10,Ang2);
a.servoWrite(11,Ang31);
XA3=XR+L1*cosd(Ang3);
YA3=YR+L1*sind(Ang3);
hold on;
plot(handles.axes5,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes5,[XA1 XA],[YA1 YA],'r-','linewidth',2);

```

```

plot(handles.axes5,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes5,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes5,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes5,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes5,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes5,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes5,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes5,Vx(Cont),Vy(Cont),'ko','linewidth',2);
hold on;
set(handles.axes5,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
title('LINEA RECTA')
xlabel('X')
ylabel('Y')
if Cont>=2
plot(handles.axes5,[Vx(i) Vx(i-1)],[Vy(i) Vy(i-1)],'c-
','linewidth',2);
hold on;
end
pause(2);
end
if M==2
e1=-(2*YA*L1);
e2=-(2*XA*L1);
e3=(XA^2)+(YA^2)+(L1^2)-(L2^2);
%Condicionante para mensaje de error, si el robot no puede
llegar graficara
%hasta la ultima posicion que pudo alcanzar
if (e3^2)>(e2^2)+(e1^2)
hold on;
set(handles.text21,'String','El robot no puede alcanzar
la posicion deseada')
break
end
%Calculo del angulo theta 1
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
set(handles.edit10,'string',Ang1);
XA1=L1*cosd(Ang1);
YA1=L1*sind(Ang1);
h=5;
XQ=30;
e4=-(2*YA*L1)-(2*L1*h*sind(phi));
e5=-(2*XA*L1)-(2*L1*h*cosd(phi))+(2*XQ*L1);
e6=(XA^2)+(YA^2)-(2*XA*XQ)+(XQ^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi))+(2*YA*h*sind(phi))-
(2*XQ*h*cosd(phi));
if (e6^2)>(e5^2)+(e4^2)
hold on;

```

```

        set(handles.text21, 'String', 'El robot no puede alcanzar
la posicion deseada')
        break
        %return
end
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
set(handles.edit11, 'string', Ang2);
XA2=XQ+L1*cosd(Ang2);
YA2=L1*sind(Ang2);
YR=25.98;
XR=15;
e7=(-2*YA*L1)+(2*YR*L1)-(2*L1*h*sind(phi+60));
e8=(-2*XA*L1)+(2*XR*L1)-(2*L1*h*cosd(phi+60));
e9=(XA^2)+(YA^2)-(2*XA*XR)-
(2*YA*YR)+(XR^2)+(YR^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi+60))+(2*YA*h*sind(phi+60))-
(2*XR*h*cosd(phi+60))-(2*YR*h*sind(phi+60));
if (e9^2)>(e7^2)+(e8^2)
    hold on;
    set(handles.text21, 'String', 'El robot no puede alcanzar
la posición deseada')
    %Detiene el programa
    %return
    break
end
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
set(handles.edit12, 'string', Ang3);
XA3=XR+L1*cosd(Ang3);
YA3=YR+L1*sind(Ang3);
hold on;
plot(handles.axes5, [0 XA1], [0 YA1], 'b-', 'linewidth', 2);
plot(handles.axes5, [XA1 XA], [YA1 YA], 'r-', 'linewidth', 2);
plot(handles.axes5, [XQ XA2], [0 YA2], 'b-', 'linewidth', 2);
plot(handles.axes5, [XA2 XB], [YA2 YB], 'r-', 'linewidth', 2);
plot(handles.axes5, [XR XA3], [YR YA3], 'b-', 'linewidth', 2);
plot(handles.axes5, [XA3 XC], [YA3 YC], 'r-', 'linewidth', 2);
plot(handles.axes5, [XA XB], [YA YB], 'y-', 'linewidth', 2);
plot(handles.axes5, [XB XC], [YB YC], 'y-', 'linewidth', 2);
plot(handles.axes5, [XA XC], [YA YC], 'y-', 'linewidth', 2);
plot(handles.axes5, Vx(Cont), Vy(Cont), 'ko', 'linewidth', 2);
hold on;
set(handles.axes5, 'xgrid', 'on', 'ygrid', 'on', 'Xlim', [-
5, 35], 'Ylim', [-10, 30])
title('LINEA RECTA')
xlabel('X')
ylabel('Y')
if Cont>=2
plot(handles.axes5, [Vx(i)    Vx(i-1)], [Vy(i)    Vy(i-1)], 'c-
', 'linewidth', 2);

```

```

hold on;
end

pause(2);
end

end
axes(handles.axes6);
cla;
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],'
Ylim',[0,30])%Delimitacion de las coordenadas del eje
w=length(Vx)-1;
for j=1:w

plot(handles.axes6,Vx(1),Vy(1),'ko','linewidth',2);
plot(handles.axes6,Vx(length(Vx)),Vy(length(Vy)),'ko','line
width',2);
plot(handles.axes6,Vx(j),Vy(j),'ko','linewidth',2);
plot(handles.axes6,[Vx(j) Vx(j+1)],[Vy(j) Vy(j+1)],'c-
','linewidth',3);
title('RECORRIDO DEL ROBOT')
xlabel('X')
ylabel('Y')
hold on;
end
XA1=[];
YA1=[];
%Obtención de las longitudes de los eslabones
L1=11.5%str2double(get(handles.A,'string'));
L2=11.5%str2double(get(handles.B,'string'));
Cont=0;
for theta=9:51%Intervalo y grado maximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes6,XA1(Cont),YA1(Cont),'r.','linewidth',2)

hold on
end
%Ubicación del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];
YB1=[];
Cont1=0;

```

```

for theta2=129:171%Intervalo y grado maximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]
plot(handles.axes6,XB1(Cont1),YB1(Cont1),'g.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],'
Ylim',[0,30])
hold on
end
%Ubicacion del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=249:291 %Intervalo y grado maximo del angulo
theta 3
Cont2=Cont2+1;
XC=XR+(L1*cosd(theta3))+(L2*cosd(theta3));
YC=YR+(L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes6,XC1(Cont2),YC1(Cont2),'b.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
hold on
end

```

BOTON LINEA RECTA CONTINUA

```

function pushbutton13_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton13 (see GCBO)
% eventdata    reserved - to be defined in a future version
of MATLAB
% handles      structure with handles and user data (see
GUIDATA)
%Rutina para la linea recta por pasos
%Seleccion del eje donde se graficara
axes(handles.axes7);
set(handles.text21,'String','')
%Limpieza del eje antes de graficar
cla;
%Activar visualización de los ángulos
set(handles.edit10,'Visible','on')
set(handles.edit11,'Visible','on')

```



```

set(handles.edit12, 'Visible', 'on')
%Obtención de las longitudes de los eslabones, coordenadas
y ángulo phi de
%la interfaz
L1=str2double(get(handles.A, 'string'));
L2=str2double(get(handles.B, 'string'));
Lim=str2double(get(handles.Linea, 'string'));
Efy=str2double(get(handles.YG, 'string'));
phi=str2double(get(handles.FI, 'string'));
%Obtención del tipo de configuración requerida para el
robot
M=str2double(get(handles.text22, 'string'));
h=5;
Cont=0;
YA=Efy-2.88*sind(30+phi)
set(handles.text21, 'String', '')
install_arduino
a=arduino('COM4');
%Configuración de pines PWM del arduino para enviar los
ángulos a los servos
a.servoAttach(9);
a.servoAttach(10);
a.servoAttach(11);
%Creación de vectores para almacenamiento de coordenadas X
y Y
Vx=[];
Vy=[];
%Almacenamiento de las coordenadas iniciales en el vector
VX y VY
%Vx=[Vx Efx];
Vy=[Vy Efy];
for i=1:0.08:Lim % Intervalo y punto máximo de X
Cont=Cont+1;
Efx=str2double(get(handles.XG, 'string'));
Efx=Efx-i;%Movimiento de derecha a izquierda
XA=Efx-2.88*cosd(30+phi);%Obtención de coordenada en X
inicial
%Almacenamiento de coordenadas
Vx=[Vx Efx]
Vy=[Vy Efy]
XB=XA+(h*cos(phi));
YB=YA+(h*sind(phi));
XC=XA+(h*cosd(phi+(60)));
YC=YA+(h*sind(phi+(60)));
%Condicionante para la primera configuración del robot
if M==1
e1=-(2*YA*L1);
e2=-(2*XA*L1);
e3=(XA^2)+(YA^2)+(L1^2)-(L2^2);

```

```

%Condicionante para mensaje de error, si el robot no puede
llegar graficara
%hasta la última posición que pudo alcanzar
if (e3^2)>(e2^2)+(e1^2)
    hold on;
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    break
end
%Cálculo del ángulo theta 1
Ang1=2*(atand((-e1)+sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
if Ang1<0
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2))
Ang1=floor(Ang1);
end
if Ang1>180
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2))
Ang1=floor(Ang1);
end
set(handles.edit10,'string',Ang1);
XA1=L1*cosd(Ang1);
YA1=L1*sind(Ang1);
h=5;
XQ=30;
e4=- (2*YA*L1) - (2*L1*h*sind(phi));
e5=- (2*XA*L1) - (2*L1*h*cosd(phi)) + (2*XQ*L1);
e6=(XA^2)+(YA^2) - (2*XA*XQ) + (XQ^2) + (h^2) + (L1^2) -
(L2^2) + (2*XA*h*cosd(phi)) + (2*YA*h*sind(phi)) -
(2*XQ*h*cosd(phi));
if (e6^2)>(e5^2)+(e4^2)
    hold on;
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    break
    %return
end
Ang2=2*(atand((-e4)+sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
if Ang2<0
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5))
Ang2=floor(Ang2);
end
if Ang2>180
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5))
Ang2=floor(Ang2);
end
set(handles.edit11,'string',Ang2);
XA2=XQ+L1*cosd(Ang2);
YA2=L1*sind(Ang2);
YR=25.98;

```

```

XR=15;
e7=(-2*YA*L1)+(2*YR*L1)-(2*L1*h*sind(phi+60));
e8=(-2*XA*L1)+(2*XR*L1)-(2*L1*h*cosd(phi+60));
e9=(XA^2)+(YA^2)-(2*XA*XR)-
(2*YA*YR)+(XR^2)+(YR^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi+60)+(2*YA*h*sind(phi+60))-
(2*XR*h*cosd(phi+60))-(2*YR*h*sind(phi+60)));
if (e9^2)>(e7^2)+(e8^2)
    hold on;
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    %Detiene el programa
    break
    %continue
end
Ang3=2*(atand((-e7)+sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
Ang31=180+(2*(atand((-e7)+sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);
if Ang3<0
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8))
Ang31=180+(2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);
end
if Ang3>180
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8))
Ang31=180+(2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);
end
set(handles.edit12,'string',Ang3);
a.servoWrite(9,Ang1);
a.servoWrite(10,Ang2);
a.servoWrite(11,Ang31);
XA3=XR+L1*cosd(Ang3);
YA3=YR+L1*sind(Ang3);
pause(0.001)
cla;
hold on;
plot(handles.axes7,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes7,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes7,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes7,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes7,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes7,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes7,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes7,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes7,[XA XC],[YA YC],'y-','linewidth',2);

```

```

plot(handles.axes7,Vx(Cont),Vy(Cont),'ko','linewidth',2);

set(handles.axes7,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
title('LINEA CONTINUA')
xlabel('X')
ylabel('Y')
pause(0.1);
end
if M==2
e1=-(2*YA*L1);
e2=-(2*XA*L1);
e3=(XA^2)+(YA^2)+(L1^2)-(L2^2);
%Condicionante para mensaje de error, si el robot no puede
llegar graficara
%hasta la última posición que pudo alcanzar
if (e3^2)>(e2^2)+(e1^2)
    hold on;
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    break
end
%Cálculo del ángulo theta 1
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
set(handles.edit10,'string',Ang1);
XA1=L1*cosd(Ang1);
YA1=L1*sind(Ang1);
h=5;
XQ=30;
e4=-(2*YA*L1)-(2*L1*h*sind(phi));
e5=-(2*XA*L1)-(2*L1*h*cosd(phi))+(2*XQ*L1);
e6=(XA^2)+(YA^2)-(2*XA*XQ)+(XQ^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi))+(2*YA*h*sind(phi))-
(2*XQ*h*cosd(phi));
if (e6^2)>(e5^2)+(e4^2)
    hold on;
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    break
end
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5))
set(handles.edit11,'string',Ang2);
XA2=XQ+L1*cosd(Ang2);
YA2=L1*sind(Ang2);
YR=25.98;
XR=15;
e7=(-2*YA*L1)+(2*YR*L1)-(2*L1*h*sind(phi+60));
e8=(-2*XA*L1)+(2*XR*L1)-(2*L1*h*cosd(phi+60));

```

```

e9=(XA^2)+(YA^2)-(2*XA*XR)-(
(2*YA*YR)+(XR^2)+(YR^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi+60))+(2*YA*h*sind(phi+60))-
(2*XR*h*cosd(phi+60))-(2*YR*h*sind(phi+60));
if (e9^2)>(e7^2)+(e8^2)
    hold on;
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    %Detiene el programa
    %return
    break
end
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
set(handles.edit12,'string',Ang3);
XA3=XR+L1*cosd(Ang3);
YA3=YR+L1*sind(Ang3);
pause(0.001)
cla;
hold on;
plot(handles.axes7,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes7,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes7,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes7,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes7,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes7,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes7,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes7,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes7,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes7,Vx(Cont),Vy(Cont),'ko','linewidth',2);

set(handles.axes7,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
title('LINEA CONTINUA')
xlabel('X')
ylabel('Y')
pause(0.1);
end

end
axes(handles.axes6);
cla;
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],
'Ylim',[0,30])%Delimitacion de las coordenadas del eje
w=length(Vx)-1;
for j=1:w
plot(handles.axes6,[Vx(j) Vx(j+1)],[Vy(j) Vy(j+1)],'c-
','linewidth',3);
hold on;
end

```

```

XA1=[];
YA1=[];
%Obtencion de las longitudes de los eslabones
L1=11.5%str2double(get(handles.A,'string'));
L2=11.5%str2double(get(handles.B,'string'));
Cont=0;
for theta=9:51%Intervalo y grado maximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes6,XA1(Cont),YA1(Cont),'r.','linewidth',2)

hold on
end
%Ubicacion del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];
YB1=[];
Cont1=0;
for theta2=129:171%Intervalo y grado maximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]
plot(handles.axes6,XB1(Cont1),YB1(Cont1),'g.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],'
Ylim',[0,30])
title('RECORRIDO DEL ROBOT')
xlabel('X')
ylabel('Y')
hold on
end
%Ubicación del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=249:291 %Intervalo y grado máximo del ángulo
theta 3
Cont2=Cont2+1;
XC=XR+(L1*cosd(theta3))+(L2*cosd(theta3));

```

```

YC=YR+(L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes6,XC1(Cont2),YC1(Cont2),'b.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
hold on
end

```

BOTON FIGURA PASOS

```

function pushbutton9_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton9 (see GCBO)
% eventdata    reserved - to be defined in a future version
of MATLAB
% handles      structure with handles and user data (see
GUIDATA)
% Rutina para el botón figura
% Selección del eje
axes(handles.axes4);
set(handles.text21,'String','')
set(handles.text21,'String','')
% Selección de las coordenadas de los puntos de la figura
[X1 Y1]=ginput;
% Almacenamiento de las coordenadas de la gráfica que se
quiera hacer
A=[X1 Y1]
disp([X1 Y1])
% Obtención de los datos previos para
axes(handles.axes5);
cla;
L1=str2double(get(handles.A,'string'));
L2=str2double(get(handles.B,'string'));
phi=str2double(get(handles.FI,'string'));
M=str2double(get(handles.text22,'string'));
h=5;
install_arduino
a=arduino('COM4');
% Configuración de pines PWM del arduino para enviar los
ángulos a los
% servos
a.servoAttach(9);
a.servoAttach(10);
a.servoAttach(11);
Cont=0;
% Condicionante de segunda configuración del robot
if M==1
length([X1 Y1])

```

```

for i=1:length([X1 Y1])
    Cont=Cont+1;
    axes(handles.axes5);
    Efx=A(i,1)
    XA=Efx-2.88*cosd(30+phi)
    Efy=A(i,2)
    YA=Efy-2.88*sind(30+phi)
    XB=XA+(h*cos(phi));
    YB=YA+(h*sind(phi));
    XC=XA+(h*cosd(phi+(60)));
    YC=YA+(h*sind(phi+(60)));
    e1=-(2*YA*L1);
    e2=-(2*XA*L1);
    e3=(XA^2)+(YA^2)+(L1^2)-(L2^2);
    if (e3^2)>(e2^2)+(e1^2)
        set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
        return
    end
    Ang1=2*(atand((-e1)+sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
    if Ang1<0
        Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
        Ang1=floor(Ang1);
    end
    if Ang1>180
        Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
        Ang1=floor(Ang1);
    end
    set(handles.edit10,'string',Ang1);
    XA1=L1*cosd(Ang1);
    YA1=L1*sind(Ang1);
    h=5;
    XQ=30;
    e4=-(2*YA*L1)-(2*L1*h*sind(phi));
    e5=-(2*XA*L1)-(2*L1*h*cosd(phi))+(2*XQ*L1);
    e6=(XA^2)+(YA^2)-(2*XA*XQ)+(XQ^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi))+(2*YA*h*sind(phi))-
(2*XQ*h*cosd(phi));
    if (e6^2)>(e5^2)+(e4^2)
        set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
        return
    end
    Ang2=2*(atand((-e4)+sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
    if Ang2<0
        Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
        Ang2=floor(Ang2);
    end
    if Ang2>180

```



```

Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
Ang2=floor(Ang2);
end
set(handles.edit11,'string',Ang2);
XA2=XQ+L1*cosd(Ang2);
YA2=L1*sind(Ang2);
YR=25.98;
XR=15;
e7=(-2*YA*L1)+(2*YR*L1)-(2*L1*h*sind(phi+60));
e8=(-2*XA*L1)+(2*XR*L1)-(2*L1*h*cosd(phi+60));
e9=(XA^2)+(YA^2)-(2*XA*XR)-
(2*YA*YR)+(XR^2)+(YR^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi+60)+(2*YA*h*sind(phi+60))-
(2*XR*h*cosd(phi+60))-(2*YR*h*sind(phi+60)));
if (e9^2)>(e7^2)+(e8^2)
    set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
    return
end
Ang3=2*(atand((-e7)+sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
if Ang3<0
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8))
Ang31=180+(2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);
end
if Ang3>180
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8))
Ang31=180+(2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);
end
set(handles.edit12,'string',Ang3);
a.servoWrite(9,Ang1);
a.servoWrite(10,Ang2);
a.servoWrite(11,Ang31);
XA3=XR+L1*cosd(Ang3);
YA3=YR+L1*sind(Ang3);
if length([X1 Y1])==2
plot(handles.axes5,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes5,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes5,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes5,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes5,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes5,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes5,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes5,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes5,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes5,Efx,Efy,'ko','linewidth',2);

```

```

set(handles.axes5,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
title('FIGURA')
xlabel('X')
ylabel('Y')
axes(handles.axes6);
cla;
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],
'Ylim',[0,30])%Delimitacion de las coordenadas del eje

plot(handles.axes6,Efx,Efy,'ko','linewidth',2);
hold on;
XA1=[];
YA1=[];
%Obtencion de las longitudes de los eslabones
L1=11%str2double(get(handles.A,'string'));
L2=11%str2double(get(handles.B,'string'));
Cont=0;
for theta=10:50%Intervalo y grado maximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes6,XA1(Cont),YA1(Cont),'r.','linewidth',2)

hold on
end
%Ubicación del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];
YB1=[];
Cont1=0;
for theta2=130:170%Intervalo y grado maximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]
plot(handles.axes6,XB1(Cont1),YB1(Cont1),'g.','linewidth',2)
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],
'Ylim',[0,30])
title('RECORRIDO DEL ROBOT')
xlabel('X')
ylabel('Y')

```

```

hold on
end
%Ubicación del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=250:290 %Intervalo y grado máximo del ángulo
theta 3
Cont2=Cont2+1;
XC=XR+(L1*cosd(theta3))+(L2*cosd(theta3));
YC=YR+(L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes6,XC1(Cont2),YC1(Cont2),'b.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
hold on
end
return
end
plot(handles.axes5,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes5,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes5,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes5,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes5,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes5,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes5,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes5,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes5,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes5,Efx,Efy,'ko','linewidth',2);
set(handles.axes5,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
title('FIGURA')
xlabel('X')
ylabel('Y')
hold on;
%hold off;
if Cont>=2
    hold on;
plot(handles.axes5,[Efx A(Cont-1,1)],[Efy A(Cont-1,2)],'c-
','linewidth',2);
end

pause(2);
end
axes(handles.axes6);

```

```

cla;
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],'
Ylim',[0,30])%Delimitacion de las coordenadas del eje
w=length([X1 Y1])-1;
for j=1:w

Efx=A(j,1)
Efy=A(j,2)
Efx1=A(j+1,1)
Efy1=A(j+1,2)
plot(handles.axes6,Efx,Efy,'ko','linewidth',2);
plot(handles.axes6,[Efx          Efx1],[Efy          Efy1],'c-
','linewidth',3);
hold on;
if j==w
plot(handles.axes6,A(j+1,1),A(j+1,2),'ko','linewidth',2);
end
end
XA1=[];
YA1=[];
%Obtencion de las longitudes de los eslabones
L1=11.5%str2double(get(handles.A,'string'));
L2=11.5%str2double(get(handles.B,'string'));
Cont=0;
for theta=9:51%Intervalo y grado maximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes6,XA1(Cont),YA1(Cont),'r.','linewidth',2)

hold on
end
%Ubicación del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];
YB1=[];
Cont1=0;
for theta2=129:171%Intervalo y grado máximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]

```

```

plot(handles.axes6, XB1 (Cont1), YB1 (Cont1), 'g.', 'linewidth', 2
)
set(handles.axes6, 'xgrid', 'on', 'ygrid', 'on', 'Xlim', [0, 35], '
Ylim', [0, 30])
title('RECORRIDO DEL ROBOT')
xlabel('X')
ylabel('Y')
hold on
end
%Ubicación del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=249:291 %Intervalo y grado máximo del ángulo
theta 3
Cont2=Cont2+1;
XC=XR+(L1*cosd(theta3))+(L2*cosd(theta3));
YC=YR+(L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes6, XC1 (Cont2), YC1 (Cont2), 'b.', 'linewidth', 2
)
set(handles.axes6, 'xgrid', 'on', 'ygrid', 'on', 'Xlim', [-
5, 35], 'Ylim', [-10, 30])
hold on
end
end
if M==2
length([X1 Y1])
for i=1:length([X1 Y1])
Cont=Cont+1;
axes(handles.axes5);
Efx=A(i,1)
XA=Efx-2.88*cosd(30+phi)
Efy=A(i,2)
YA=Efy-2.88*sind(30+phi)
XB=XA+(h*cos(phi));
YB=YA+(h*sind(phi));
XC=XA+(h*cosd(phi+(60)));
YC=YA+(h*sind(phi+(60)));
e1=-(2*YA*L1);
e2=-(2*XA*L1);
e3=(XA^2)+(YA^2)+(L1^2)-(L2^2);
if (e3^2)>(e2^2)+(e1^2)
set(handles.text21, 'String', 'El robot no puede alcanzar
la posicion deseada')
return

```

```

end
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
set(handles.edit10,'string',Ang1);
XA1=L1*cosd(Ang1);
YA1=L1*sind(Ang1);
h=5;
XQ=30;
e4=- (2*YA*L1) - (2*L1*h*sind(phi));
e5=- (2*XA*L1) - (2*L1*h*cosd(phi)) + (2*XQ*L1);
e6=(XA^2)+(YA^2) - (2*XA*XQ) + (XQ^2) + (h^2) + (L1^2) -
(L2^2) + (2*XA*h*cosd(phi)) + (2*YA*h*sind(phi)) -
(2*XQ*h*cosd(phi));
if (e6^2)>(e5^2)+(e4^2)
    set(handles.text21,'String','El robot no puede alcanzar
la posicion deseada')
    return
end
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
set(handles.edit11,'string',Ang2);
XA2=XQ+L1*cosd(Ang2);
YA2=L1*sind(Ang2);
YR=25.98;
XR=15;
e7=(-2*YA*L1) + (2*YR*L1) - (2*L1*h*sind(phi+60));
e8=(-2*XA*L1) + (2*XR*L1) - (2*L1*h*cosd(phi+60));
e9=(XA^2)+(YA^2) - (2*XA*XR) -
(2*YA*YR) + (XR^2) + (YR^2) + (h^2) + (L1^2) -
(L2^2) + (2*XA*h*cosd(phi+60)) + (2*YA*h*sind(phi+60)) -
(2*XR*h*cosd(phi+60)) - (2*YR*h*sind(phi+60));
if (e9^2)>(e7^2)+(e8^2)
    set(handles.text21,'String','El robot no puede alcanzar
la posicion deseada')
    return
end
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
set(handles.edit12,'string',Ang3);
XA3=XR+L1*cosd(Ang3);
YA3=YR+L1*sind(Ang3);
if length([X1 Y1])==2
plot(handles.axes5,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes5,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes5,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes5,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes5,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes5,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes5,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes5,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes5,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes5,Efx,Efy,'ko','linewidth',2);

```

```

set(handles.axes5,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
title('FIGURA')
xlabel('X')
ylabel('Y')
axes(handles.axes6);
cla;
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],
'Ylim',[0,30])%Delimitacion de las coordenadas del eje

plot(handles.axes6,Efx,Efy,'ko','linewidth',2);
hold on;
XA1=[];
YA1=[];
%Obtencion de las longitudes de los eslabones
L1=11.5%str2double(get(handles.A,'string'));
L2=11.5%str2double(get(handles.B,'string'));
Cont=0;
for theta=9:51%Intervalo y grado maximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes6,XA1(Cont),YA1(Cont),'r.','linewidth',2)

hold on
end
%Ubicacion del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];
YB1=[];
Cont1=0;
for theta2=129:171%Intervalo y grado maximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]
plot(handles.axes6,XB1(Cont1),YB1(Cont1),'g.','linewidth',2)
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],
'Ylim',[0,30])
title('RECORRIDO DEL ROBOT')
xlabel('X')
ylabel('Y')

```

```

hold on
end
%Ubicacion del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=249:291 %Intervalo y grado máximo del ángulo
theta 3
Cont2=Cont2+1;
XC=XR+(L1*cosd(theta3))+(L2*cosd(theta3));
YC=YR+(L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes6,XC1(Cont2),YC1(Cont2),'b.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
hold on
end
return
end
plot(handles.axes5,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes5,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes5,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes5,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes5,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes5,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes5,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes5,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes5,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes5,Efx,Efy,'ko','linewidth',2);
set(handles.axes5,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
title('FIGURA')
xlabel('X')
ylabel('Y')
hold on;
%hold off;
if Cont>=2
    hold on;
plot(handles.axes5,[Efx A(Cont-1,1)],[Efy A(Cont-1,2)],'c-
','linewidth',2);
end

pause(2);
end
axes(handles.axes6);

```



```

cla;
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],'
Ylim',[0,30])%Delimitacion de las coordenadas del eje
w=length([X1 Y1])-1;
for j=1:w

Efx=A(j,1)
Efy=A(j,2)
Efx1=A(j+1,1)
Efy1=A(j+1,2)
plot(handles.axes6,Efx,Efy,'ko','linewidth',2);
plot(handles.axes6,[Efx          Efx1],[Efy          Efy1],'c-
','linewidth',3);
hold on;
if j==w
plot(handles.axes6,A(j+1,1),A(j+1,2),'ko','linewidth',2);
end
end
XA1=[];
YA1=[];
%Obtención de las longitudes de los eslabones
L1=11.5%str2double(get(handles.A,'string'));
L2=11.5%str2double(get(handles.B,'string'));
Cont=0;
for theta=9:51%Intervalo y grado maximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes6,XA1(Cont),YA1(Cont),'r.','linewidth',2)

hold on
end
%Ubicación del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];
YB1=[];
Cont1=0;
for theta2=129:171%Intervalo y grado máximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]

```

```

plot(handles.axes6, XB1 (Cont1), YB1 (Cont1), 'g.', 'linewidth', 2
)
set(handles.axes6, 'xgrid', 'on', 'ygrid', 'on', 'Xlim', [0, 35], '
Ylim', [0, 30])
title('RECORRIDO')
xlabel('X')
ylabel('Y')
hold on
end
%Ubicacion del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=249:291 %Intervalo y grado máximo del ángulo
theta 3
Cont2=Cont2+1;
XC=XR+(L1*cosd(theta3))+(L2*cosd(theta3));
YC=YR+(L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes6, XC1 (Cont2), YC1 (Cont2), 'b.', 'linewidth', 2
)
set(handles.axes6, 'xgrid', 'on', 'ygrid', 'on', 'Xlim', [-
5, 35], 'Ylim', [-10, 30])
hold on
end
end

```

BOTON FIGURA CONTINUA

```

function pushbutton14_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton14 (see GCBO)
% eventdata    reserved - to be defined in a future version
of MATLAB
% handles      structure with handles and user data (see
GUIDATA)
% Rutina para el boton figura Continuo
% Selección del eje
axes(handles.axes7);
set(handles.text21, 'String', '')
% Selección de las coordenadas de los puntos de la figura
[X1 Y1]=ginput;
% Almacenamiento de las coordenadas de la gráfica que se
quiera hacer
A=[X1 Y1]
disp([X1 Y1])
% Obtención de los datos previos para

```

```

axes(handles.axes7);
cla;
L1=str2double(get(handles.A,'string'));
L2=str2double(get(handles.B,'string'));
phi=str2double(get(handles.FI,'string'));
M=str2double(get(handles.text22,'string'));
install_arduino
a=arduino('COM4');
%Configuración de pines PWM del arduino para enviar los
ángulos a los servos
a.servoAttach(9);
a.servoAttach(10);
a.servoAttach(11);
h=5;
Cont=0;
%Condicionante de segunda configuración del robot
if M==1
length([X1 Y1])
for i=1:0.08:length([X1 Y1])
    Cont=Cont+1;
    axes(handles.axes7);
    Efx=A(i,1)
    XA=Efx-2.88*cosd(30+phi)
    Efy=A(i,2)
    YA=Efy-2.88*sind(30+phi)
    XB=XA+(h*cos(phi));
    YB=YA+(h*sind(phi));
    XC=XA+(h*cosd(phi+(60)));
    YC=YA+(h*sind(phi+(60)));
    e1=-(2*YA*L1);
    e2=-(2*XA*L1);
    e3=(XA^2)+(YA^2)+(L1^2)-(L2^2);
    if (e3^2)>(e2^2)+(e1^2)
        set(handles.text21,'String','El robot no puede alcanzar
la posición deseada')
        return
    end
    Ang1=2*(atand((-e1)+sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
    if Ang1<0
        Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2))
        Ang1=floor(Ang1);
    end
    if Ang1>180
        Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2))
        Ang1=floor(Ang1);
    end
    set(handles.edit10,'string',Ang1);
    XA1=L1*cosd(Ang1);
    YA1=L1*sind(Ang1);

```

```

h=5;
XQ=30;
e4=- (2*YA*L1) - (2*L1*h*sind(phi));
e5=- (2*XA*L1) - (2*L1*h*cosd(phi)) + (2*XQ*L1);
e6=(XA^2) + (YA^2) - (2*XA*XQ) + (XQ^2) + (h^2) + (L1^2) -
(L2^2) + (2*XA*h*cosd(phi)) + (2*YA*h*sind(phi)) -
(2*XQ*h*cosd(phi));
if (e6^2) > (e5^2) + (e4^2)
    set(handles.text21, 'String', 'El robot no puede alcanzar
la posición deseada')
    return
end
Ang2=2* (atand((-e4)+sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
if Ang2 < 0
Ang2=2* (atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
Ang2=floor(Ang2);
end
if Ang2 > 180
Ang2=2* (atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
Ang2=floor(Ang2);
end
set(handles.edit11, 'string', Ang2);
XA2=XQ+L1*cosd(Ang2);
YA2=L1*sind(Ang2);
YR=25.98;
XR=15;
e7=(-2*YA*L1) + (2*YR*L1) - (2*L1*h*sind(phi+60));
e8=(-2*XA*L1) + (2*XR*L1) - (2*L1*h*cosd(phi+60));
e9=(XA^2) + (YA^2) - (2*XA*XR) -
(2*YA*YR) + (XR^2) + (YR^2) + (h^2) + (L1^2) -
(L2^2) + (2*XA*h*cosd(phi+60)) + (2*YA*h*sind(phi+60)) -
(2*XR*h*cosd(phi+60)) - (2*YR*h*sind(phi+60));
if (e9^2) > (e7^2) + (e8^2)
    set(handles.text21, 'String', 'El robot no puede alcanzar
la posición deseada')
    return
end
Ang3=2* (atand((-e7)+sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
if Ang3 < 0
Ang3=2* (atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
Ang31=180+(2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);
end
if Ang3 > 180
Ang3=2* (atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
Ang31=180+(2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-
e8)))
Ang31=floor(Ang31);

```

```

end
set(handles.edit12,'string',Ang3);
a.servoWrite(9,Ang1);
a.servoWrite(10,Ang2);
a.servoWrite(11,Ang31);
XA3=XR+L1*cosd(Ang3);
YA3=YR+L1*sind(Ang3);
if length([X1 Y1])==2
plot(handles.axes7,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes7,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes7,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes7,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes7,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes7,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes7,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes7,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes7,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes7,Efx,Efy,'ko','linewidth',2);
set(handles.axes7,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
axes(handles.axes7);
cla;
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],'
Ylim',[0,30])%Delimitacion de las coordenadas del eje

plot(handles.axes6,Efx,Efy,'ko','linewidth',2);
hold on;
XA1=[];
YA1=[];
%Obtencion de las longitudes de los eslabones
L1=11%str2double(get(handles.A,'string'));
L2=11%str2double(get(handles.B,'string'));
Cont=0;
for theta=10:50%Intervalo y grado maximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes6,XA1(Cont),YA1(Cont),'r.','linewidth',2)

hold on
end
%Ubicacion del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];

```

```

YB1=[];
Cont1=0;
for theta2=130:170%Intervalo y grado máximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]
plot(handles.axes6,XB1(Cont1),YB1(Cont1),'g.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],'
Ylim',[0,30])
hold on
end
%Ubicación del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=250:290 %Intervalo y grado maximo del angulo
theta 3
Cont2=Cont2+1;
XC=XR+(L1*cosd(theta3))+(L2*cosd(theta3));
YC=YR+(L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes6,XC1(Cont2),YC1(Cont2),'b.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
hold on
end
return
end
plot(handles.axes7,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes7,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes7,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes7,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes7,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes7,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes7,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes7,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes7,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes7,Efx,Efy,'ko','linewidth',2);
set(handles.axes7,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
hold on;
%hold off;

```

```

if Cont>=2
    hold on;
plot(handles.axes7,[Efx A(Cont-1,1)], [Efy A(Cont-1,2)], 'c-
', 'linewidth', 2);
end

pause(2);
end
axes(handles.axes6);
cla;
set(handles.axes6, 'xgrid', 'on', 'ygrid', 'on', 'Xlim', [0,35], '
Ylim', [0,30])%Delimitacion de las coordenadas del eje
w=length([X1 Y1])-1;
for j=1:w

Efx=A(j,1)
Efy=A(j,2)
Efx1=A(j+1,1)
Efy1=A(j+1,2)
plot(handles.axes6,Efx,Efy, 'ko', 'linewidth', 2);
plot(handles.axes6, [Efx          Efx1], [Efy          Efy1], 'c-
', 'linewidth', 3);
hold on;
if j==w
plot(handles.axes6,A(j+1,1),A(j+1,2), 'ko', 'linewidth', 2);
end
end
XA1=[];
YA1=[];
%Obtencion de las longitudes de los eslabones
L1=11.5%str2double(get(handles.A, 'string'));
L2=11.5%str2double(get(handles.B, 'string'));
Cont=0;
for theta=9:51%Intervalo y grado maximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes6,XA1(Cont), YA1(Cont), 'r.', 'linewidth', 2)

hold on
end
%Ubicacion del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];

```

```

YB1=[];
Cont1=0;
for theta2=129:171%Intervalo y grado maximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]
plot(handles.axes6,XB1(Cont1),YB1(Cont1),'g.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],'
Ylim',[0,30])
hold on
end
%Ubicacion del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=249:291 %Intervalo y grado máximo del ángulo
theta 3
Cont2=Cont2+1;
XC=XR+(L1*cosd(theta3))+(L2*cosd(theta3));
YC=YR+(L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes6,XC1(Cont2),YC1(Cont2),'b.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
hold on
end
end
if M==2
length([X1 Y1])
for i=1:length([X1 Y1])
Cont=Cont+1;
axes(handles.axes5);
Efx=A(i,1)
XA=Efx-2.88*cosd(30+phi)
Efy=A(i,2)
YA=Efy-2.88*sind(30+phi)
XB=XA+(h*cos(phi));
YB=YA+(h*sind(phi));
XC=XA+(h*cosd(phi+(60)));
YC=YA+(h*sind(phi+(60)));
e1=-(2*YA*L1);
e2=-(2*XA*L1);

```



```

e3=(XA^2)+(YA^2)+(L1^2)-(L2^2);
if (e3^2)>(e2^2)+(e1^2)
    set(handles.text21,'String','El robot no puede alcanzar
la posicion deseada')
    return
end
Ang1=2*(atand((-e1)-sqrt((e1^2)+(e2^2)-(e3^2)))/(e3-e2));
set(handles.edit10,'string',Ang1);
XA1=L1*cosd(Ang1);
YA1=L1*sind(Ang1);
h=5;
XQ=30;
e4=- (2*YA*L1) - (2*L1*h*sind(phi));
e5=- (2*XA*L1) - (2*L1*h*cosd(phi)) + (2*XQ*L1);
e6=(XA^2)+(YA^2)-(2*XA*XQ)+(XQ^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi))+(2*YA*h*sind(phi))-
(2*XQ*h*cosd(phi));
if (e6^2)>(e5^2)+(e4^2)
    set(handles.text21,'String','El robot no puede alcanzar
la posicion deseada')
    return
end
Ang2=2*(atand((-e4)-sqrt((e4^2)+(e5^2)-(e6^2)))/(e6-e5));
set(handles.edit11,'string',Ang2);
XA2=XQ+L1*cosd(Ang2);
YA2=L1*sind(Ang2);
YR=25.98;
XR=15;
e7=(-2*YA*L1)+(2*YR*L1)-(2*L1*h*sind(phi+60));
e8=(-2*XA*L1)+(2*XR*L1)-(2*L1*h*cosd(phi+60));
e9=(XA^2)+(YA^2)-(2*XA*XR)-
(2*YA*YR)+(XR^2)+(YR^2)+(h^2)+(L1^2)-
(L2^2)+(2*XA*h*cosd(phi+60))+(2*YA*h*sind(phi+60))-
(2*XR*h*cosd(phi+60))-(2*YR*h*sind(phi+60));
if (e9^2)>(e7^2)+(e8^2)
    set(handles.text21,'String','El robot no puede alcanzar
la posicion deseada')
    return
end
Ang3=2*(atand((-e7)-sqrt((e7^2)+(e8^2)-(e9^2)))/(e9-e8));
set(handles.edit12,'string',Ang3);
XA3=XR+L1*cosd(Ang3);
YA3=YR+L1*sind(Ang3);
if length([X1 Y1])==2
plot(handles.axes7,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes7,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes7,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes7,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes7,[XR XA3],[YR YA3],'b-','linewidth',2);

```

```

plot(handles.axes7,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes7,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes7,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes7,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes7,Efx,Efy,'ko','linewidth',2);
set(handles.axes7,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
axes(handles.axes7);
cla;
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],
'Ylim',[0,30])%Delimitacion de las coordenadas del eje

plot(handles.axes6,Efx,Efy,'ko','linewidth',2);
hold on;
XA1=[];
YA1=[];
%Obtencion de las longitudes de los eslabones
L1=11.5%str2double(get(handles.A,'string'));
L2=11.5%str2double(get(handles.B,'string'));
Cont=0;
for theta=9:51%Intervalo y grado maximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes6,XA1(Cont),YA1(Cont),'r.','linewidth',2)

hold on
end
%Ubicacion del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];
YB1=[];
Cont1=0;
for theta2=129:171%Intervalo y grado maximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]
plot(handles.axes6,XB1(Cont1),YB1(Cont1),'g.','linewidth',2)
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],
'Ylim',[0,30])
hold on

```

```

end
%Ubicación del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=249:291 %Intervalo y grado máximo del ángulo
theta 3
Cont2=Cont2+1;
XC=XR+(L1*cosd(theta3))+(L2*cosd(theta3));
YC=YR+(L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes6,XC1(Cont2),YC1(Cont2),'b.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
hold on
end
return
end
plot(handles.axes7,[0 XA1],[0 YA1],'b-','linewidth',2);
plot(handles.axes7,[XA1 XA],[YA1 YA],'r-','linewidth',2);
plot(handles.axes7,[XQ XA2],[0 YA2],'b-','linewidth',2);
plot(handles.axes7,[XA2 XB],[YA2 YB],'r-','linewidth',2);
plot(handles.axes7,[XR XA3],[YR YA3],'b-','linewidth',2);
plot(handles.axes7,[XA3 XC],[YA3 YC],'r-','linewidth',2);
plot(handles.axes7,[XA XB],[YA YB],'y-','linewidth',2);
plot(handles.axes7,[XB XC],[YB YC],'y-','linewidth',2);
plot(handles.axes7,[XA XC],[YA YC],'y-','linewidth',2);
plot(handles.axes7,Efx,Efy,'ko','linewidth',2);
set(handles.axes7,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
hold on;
%hold off;
if Cont>=2
    hold on;
plot(handles.axes7,[Efx A(Cont-1,1)],[Efy A(Cont-1,2)],'c-
','linewidth',2);
end

pause(2);
end
axes(handles.axes6);
cla;
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],
'Ylim',[0,30])%Delimitacion de las coordenadas del eje
w=length([X1 Y1])-1;

```

```

for j=1:w

Efx=A(j,1)
Efy=A(j,2)
Efx1=A(j+1,1)
Efy1=A(j+1,2)
plot(handles.axes6,Efx,Efy,'ko','linewidth',2);
plot(handles.axes6,[Efx      Efx1],[Efy      Efy1],'c-
','linewidth',3);
hold on;
if j==w
plot(handles.axes6,A(j+1,1),A(j+1,2),'ko','linewidth',2);
end
end
XA1=[];
YA1=[];
%Obtención de las longitudes de los eslabones
L1=11.5%str2double(get(handles.A,'string'));
L2=11.5%str2double(get(handles.B,'string'));
Cont=0;
for theta=9:51%Intervalo y grado máximo para theta 1
Cont=Cont+1;
XA=(L1*cosd(theta))+(L2*cosd(theta));%Coordenada del punto
A en X
YA=(L1*sind(theta))+(L2*sind(theta));%Coordenada del punto
A en Y
XA1=[XA1 XA]%Almacenamiento de coordenadas A en X
YA1=[YA1 YA]%Almacenamiento de coordenadas A en Y
%Grafica del punto A
plot(handles.axes6,XA1(Cont),YA1(Cont),'r.','linewidth',2)

hold on
end
%Ubicación del punto Q que es (XQ,XY)=(30,0)
XQ=30;
XB1=[];
YB1=[];
Cont1=0;
for theta2=129:171%Intervalo y grado máximo para theta 2
Cont1=Cont1+1;
XB=XQ+(L1*cosd(theta2))+(L2*cosd(theta2));
YB=(L1*sind(theta2))+(L2*sind(theta2));
XB1=[XB1 XB]
YB1=[YB1 YB]
plot(handles.axes6,XB1(Cont1),YB1(Cont1),'g.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[0,35],'
Ylim',[0,30])
hold on

```

```

end
%Ubicación del punto Q que es (XQ,XY)=(15,25.8)
XR=15;
YR=25.8;
XC1=[];
YC1=[];
Cont2=0;
for theta3=249:291 %Intervalo y grado máximo del ángulo
theta 3
Cont2=Cont2+1;
XC=XR+(L1*cosd(theta3))+(L2*cosd(theta3));
YC=YR+(L1*sind(theta3))+(L2*sind(theta3));
XC1=[XC1 XC]
YC1=[YC1 YC]
plot(handles.axes6,XC1(Cont2),YC1(Cont2),'b.','linewidth',2
)
set(handles.axes6,'xgrid','on','ygrid','on','Xlim',[-
5,35],'Ylim',[-10,30])
hold on
end
end

```

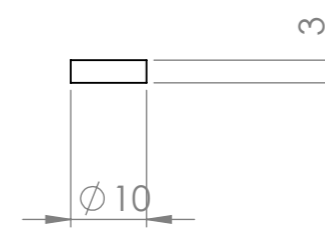
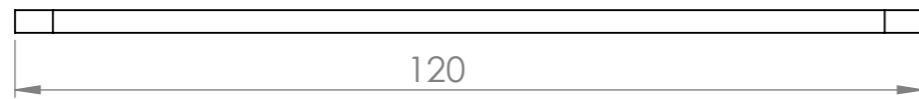
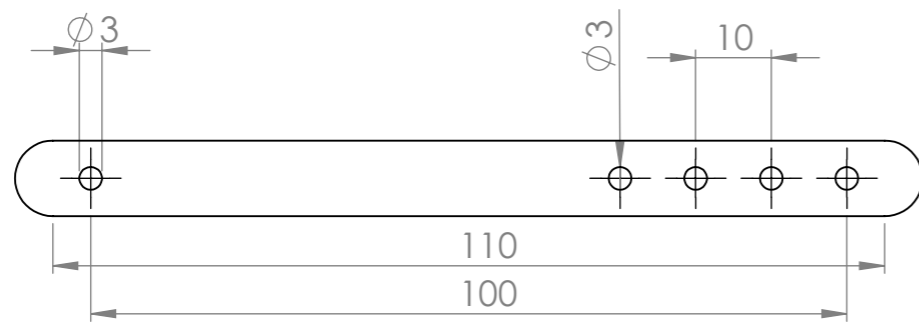
BOTON ORIGEN

```

function pushbutton11_Callback(hObject, eventdata, handles)
%Rutina para poner a 0° los servomotores
%Definición de parámetros de arduino
clc;
clear all;
close all;
set(handles.text21,'String','')
install_arduino
a=arduino('COM4');
a.servoAttach(9);
a.servoAttach(10);
a.servoAttach(11);
%Envío de datos a los servomotores
a.servoWrite(9,0);
a.servoWrite(10,0);
a.servoWrite(11,0);

```

ANEXO F



NOMBRES
YORDY TORRES
JUAN TOVAR

APROBACIÓN

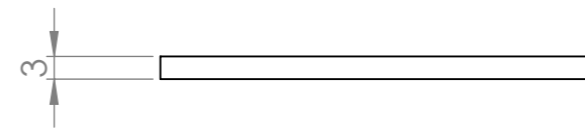
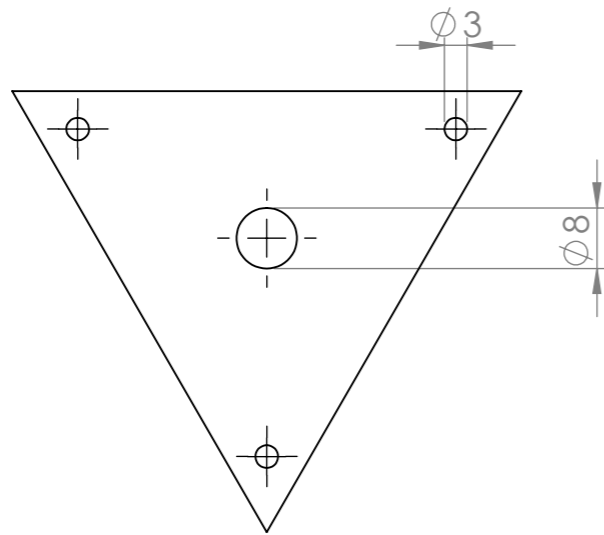
N.º DE DIBUJO

Eslabon b

A3

ESCALA: 1:1

HOJA 1 DE 1



NOMBRES
YORDY
JUAN

APELLIDOS
TORRES
TOVAR

APROBACIÓN

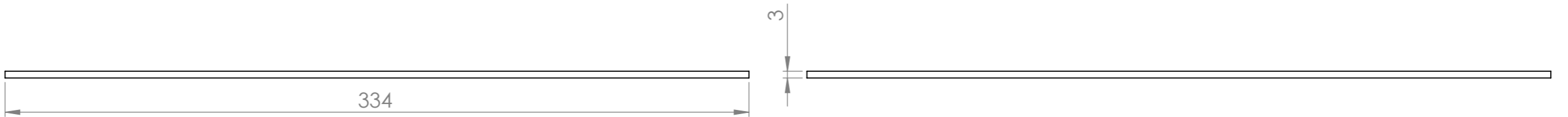
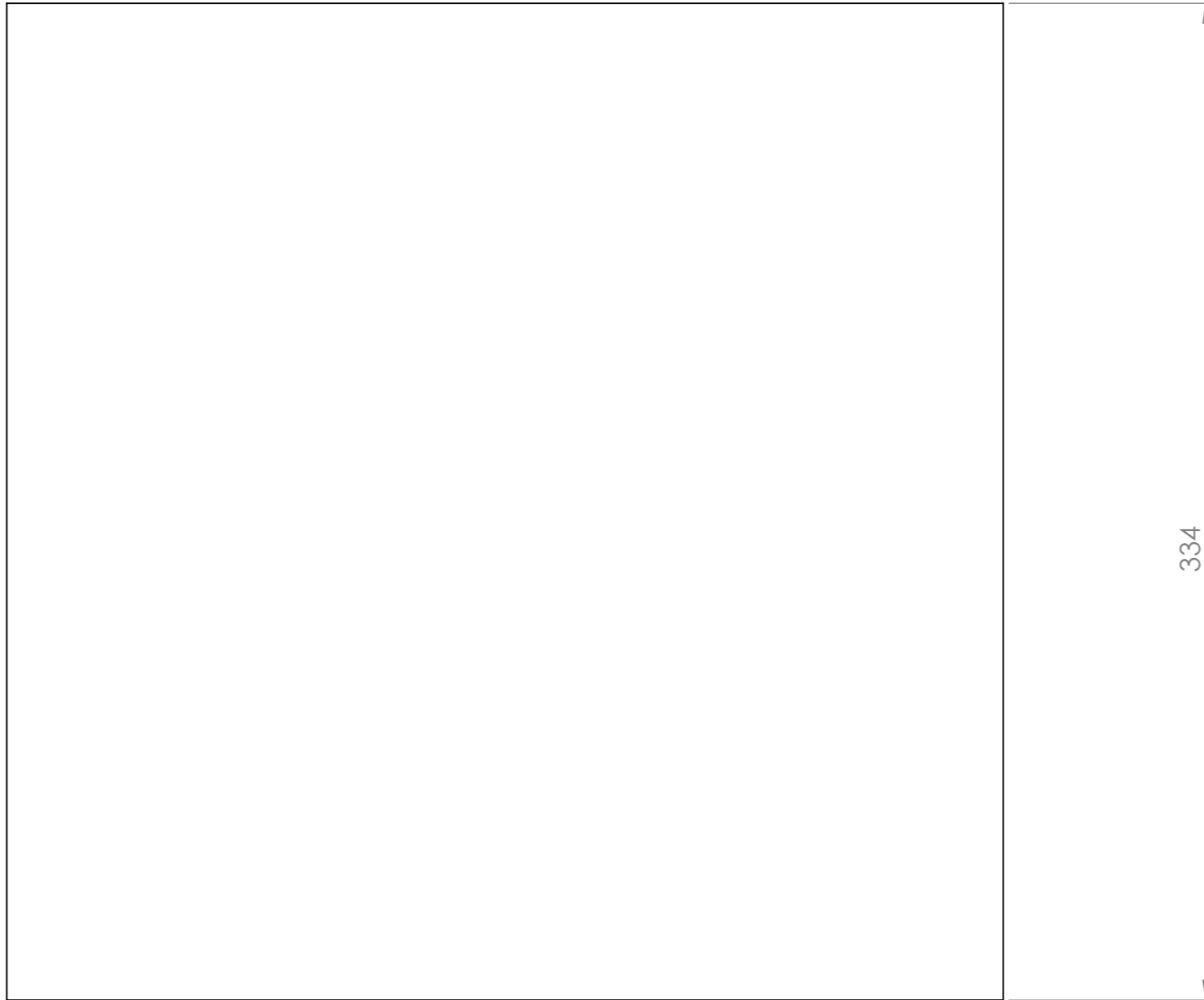
N.º DE DIBUJO

P. MOVIL

A3

ESCALA: 1:1

HOJA 1 DE 1



NOMBRES
YORDY TORRES
JUAN TOVAR

APROBACIÓN

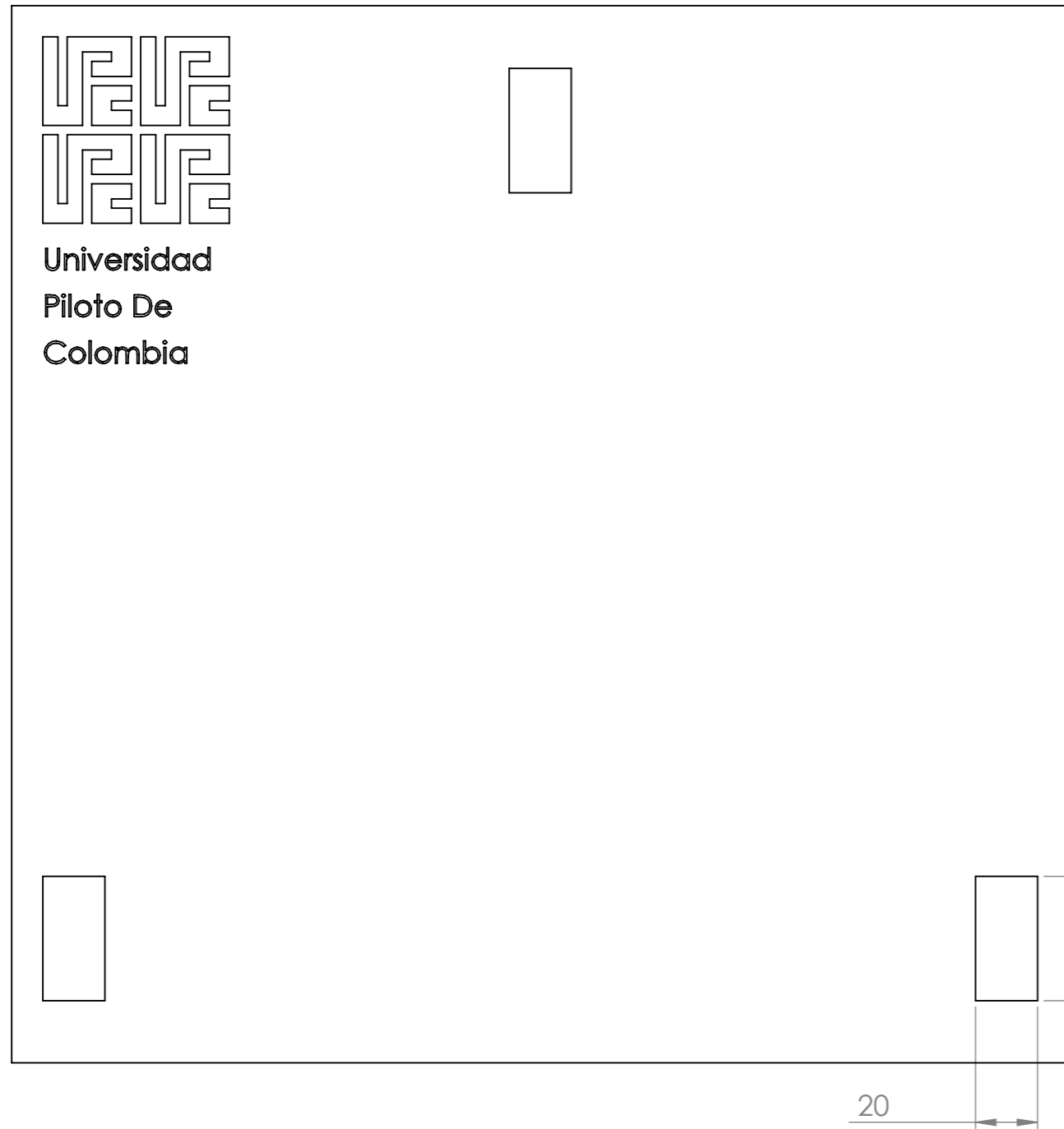
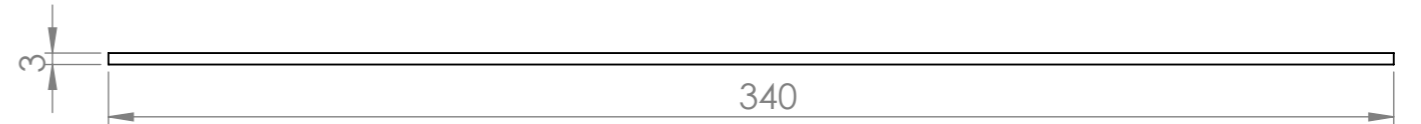
N.º DE DIBUJO

base

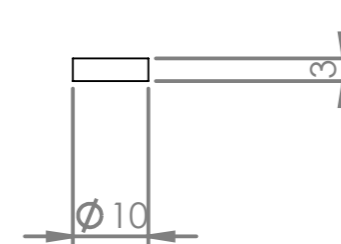
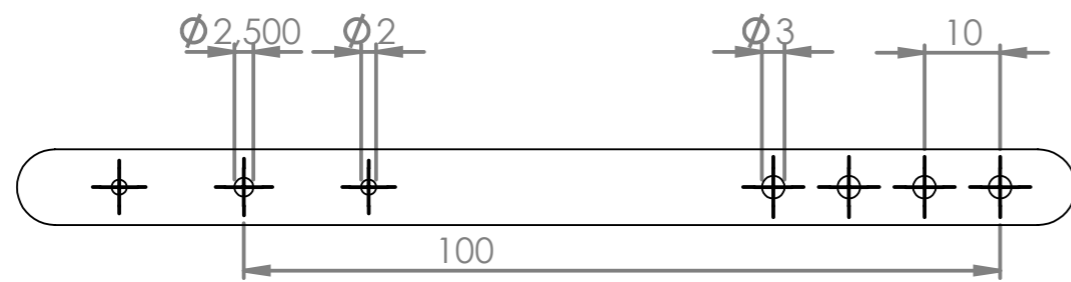
A3

ESCALA: 1:1

HOJA 1 DE 1



NOMBRES YORDY JUAN	APELLIDOS TORRES TOVAR	APROBACIÓN	N.º DE DIBUJO BASE LOGO	A3
ESCALA: 1:2			HOJA 1 DE 1	



NOMBRES
YORDY TORRES
JUAN TOVAR

APROBACIÓN

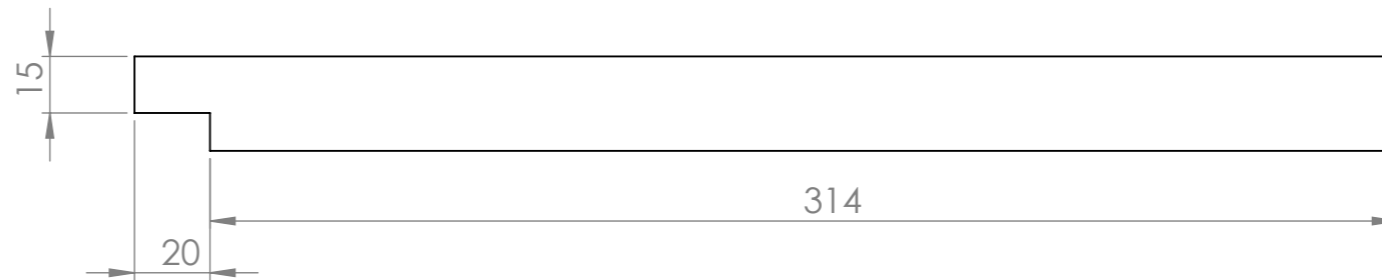
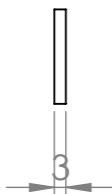
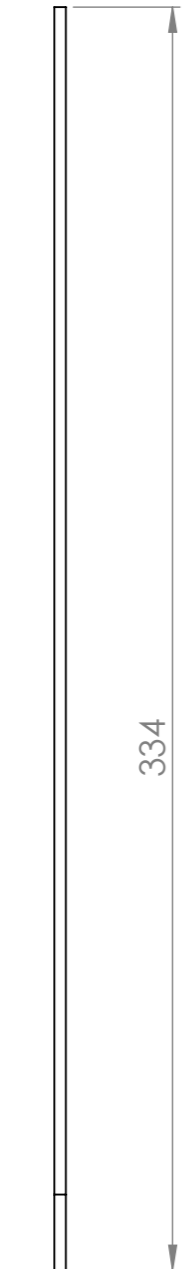
N.º DE DIBUJO

Eslabon a

A3

ESCALA: 1:1

HOJA 1 DE 1



NOMBRES APELLIDOS
YORDY TORRES
JUAN TOVAR

APROBACIÓN

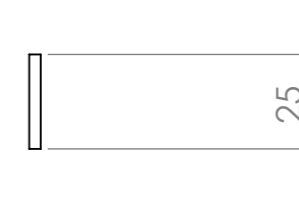
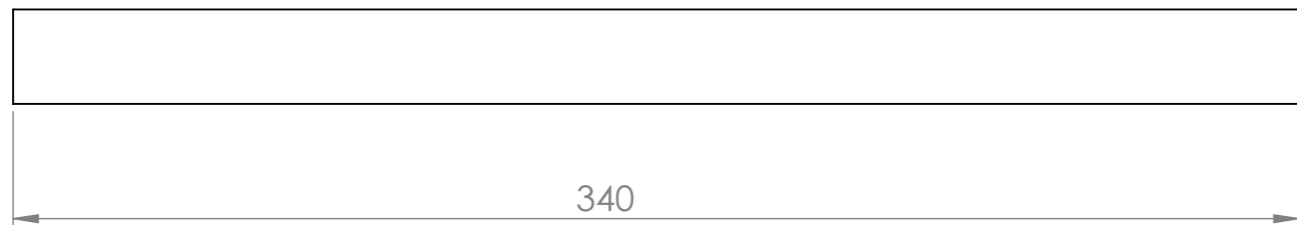
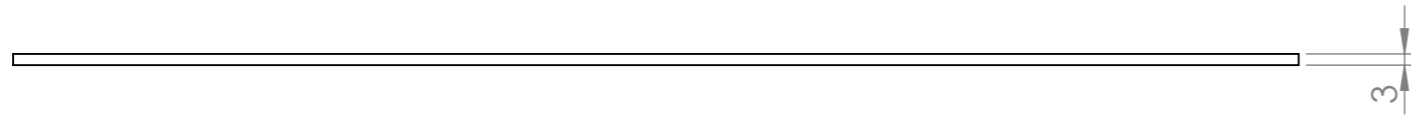
N.º DE DIBUJO

UNION LADOS

A3

ESCALA: 1:5

HOJA 1 DE 1



NOMBRES
YORDY TORRES
JUAN TOVAR

APROBACIÓN

N.º DE DIBUJO

UNION FRENTE

A3

ESCALA: 1:2

HOJA 1 DE 1

